

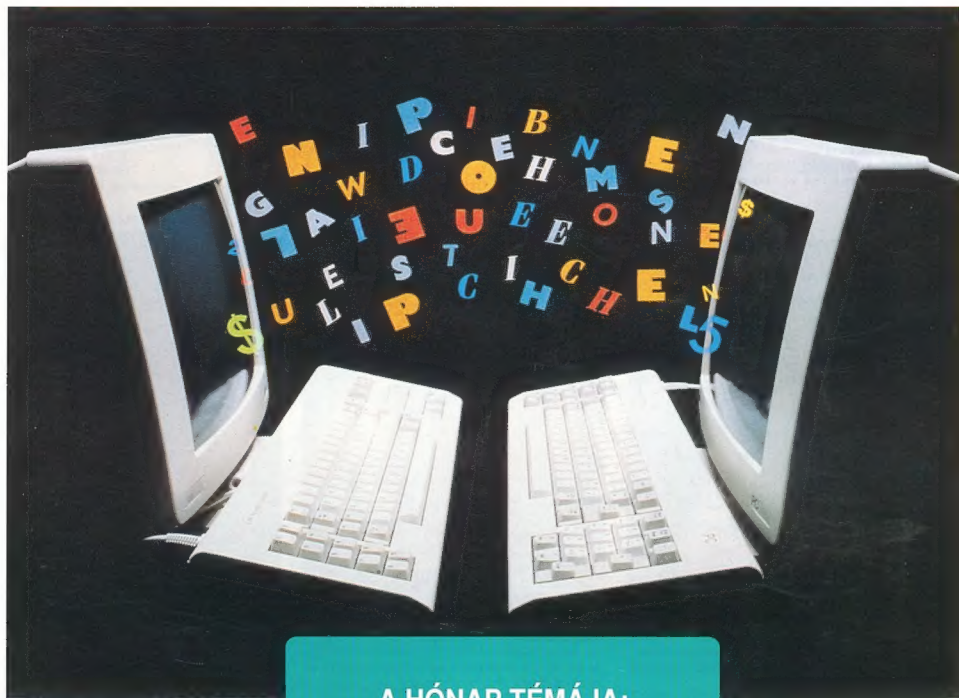
1992 / AUGUSZTUS

ÁRA: 196 FT

ALAPLAP



MIKROSZÁMÍTÓGÉP MAGAZIN MÁGNESLEMEZ MELLÉKLETTEL



A HÓNAP TÉMÁJA:

SZÓCSÉPLÉS?

Gépírók a mennyben

Csaknem Basic

Mi muzsikusz lelkek...

Számkígyó-bűvölés

Volt egyszer egy PC Turbo Klub

Intelligencia minden szinten

Taszkkommunikációs eszközök

A hardver(es) jövője

A CorelDRAW méltó vetélytársa

Átlépés egy másik dimenzióba

A MÁGNESLEMEZEN:

Komplex számolás
„Biliárdozó marslakók”
Tilitoli Herculesre is
Szóhossz-statisztika
Számfelismerő



1. Támasztékos

Könnyű, könnyedén mozgatható. A legolcsóbb modell. (Ezen nincs digitális óra.)



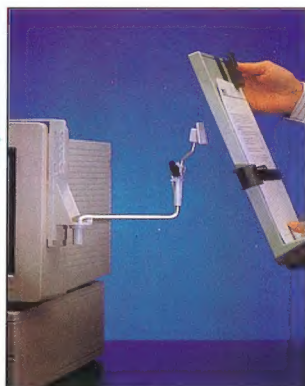
2. Talpas

A legnépszerűbb változat. Bármilyen asztalfelületen könnyedén, kis helyen lerakható, máshová bármikor áttehető.



3. Alátétes

Kis helyet foglal el. A számlítógép, a monitor vagy az írógép alá csúsztatott támaszték hordozza az írártatót táblát.



4. Felragasztható

Kétoldalas tapadószalaggal rögzíthető a monitor oldalára. Asztalfelületet nem igényel. Használata-
n kívül a tábla leemelhető róla.



5. Hozsúkarú

Nagy teherbírású, sokféleképpen beállítható, az asztalra szorítócsavarral lefoglalható.

Általános szolgáltatások:

- Beépített digitális órával felszerelt papírcsújtató.
- A kéziratok vastagságához igazodó, átlátszó sorvezető.
- A nagyobb méretű eredetikek is rögzítő szerkezet.
- Állítható dőlésszögű tábla.

Az igényesebbeknek néhány extra lehetőség:

- Lábpedálos működtetésű, motorizált sorvezető. Ezáltal a gépelés megszakítása nélkül, folyamatosan követhető az olvasott szöveg.
- Olvasólámpa a táblára erősítve. Különösen a nehezen olvasható kéziratokhoz hasznos.
- Nagyító szerkezet az apróbetűs szövegekhez. Egyszerre két normál sornyi tekinthető át vele.
- Táblatoldalék a szélesebb kéziratokhoz.



ALAPLAP

Mikroszámítógép magazin
mágneslemez melléklettel

Megjelenik havonta

Főszerkesztő:
Faklen Pál

Főszerkesztő-helyettes:
Varga János

Szerkesztő:
Jakab Ágnes

Munkatárs:
Sziebig Andrea

A mágneslemez melléklet,
a Lemezkalauz
és a Közkincs szerkesztője:
Verebély Pálné

A szerkesztőbizottság tagjai:
Barna László, Boros György,
Broczkó Péter, Brüll Károly,
Farkas Ernő, Feleki Zoltán,
Herczeg József, Kassay Árpád,
Kónya László, Kovács P. Attila,
Pintér Gábor, Vargha Dénes,
Vékony Tamás, Villányi László,
Zoltai Péter

Szerkesztőség, kiadó
és hirdetésszervezés:

1441 Budapest
VIII., Reguly Antal u. 8.
Telefon és fax: 133-1839

Felelős kiadó:
Sebestyén Ilona
ügyvezető igazgató



Cédus Kiadó Kft

Nyomdai előkészítés:
Tipoprint Kft, Budapest

Nyomtatás:
Zalai Nyomda, Zalaegerszeg
Felelős vezető: Galla József

Terjeszti a Magyar Posta.
Előfizethető a hírlapkézbesítő
postahivataloknál és a Posta
Hírlapelőfizetési és Lapellátási
Irodájánál (XIII., Lehel u. 10/a,
Budapest 1900), vagy átutalással
a 215-96162 pénzforgalmi számmal.

Példányonkénti ár: 196 Ft
Évi előfizetési díj: 2 352 Ft

Külföldre terjeszti a Kultúra,
Pf. 149, Budapest 1389

HU ISSN 0865-9788

A HÓNAP TÉMÁJA: SZÖCSÉPLÉS?

- 2 Az egész szakma szerszáma
- 3 A tűzcsapból is víz folyik
(Farkas Ernő)
- 5 Angol-magyar mérkőzés
(Seregy Lajos)
- 7 Olvass.el (!)
(Faklen Pál—Varga János)
- 9 Szövegekben böngészve
(Farkas Ernő)
- 10 Helyesírási gerillaháború
(Faklen Pál)
- 12 Multimédia vagy multimédium ?
(Faklen Pál)
- 14 100 szónak mennyi (lesz) a vége ?

14 TÉMAVÖITŐ

TUDÁSTECHNOLÓGIA

- 17 A tudás minden előtt!
(Sántáné Tóth Edit)

GÉPRAJZ

- 23 Vége a mellőzöttségnek
(Lóth Tamás)

KÖZKINC

- 26 Csaknem Basic (Sándor András)
- 27 Mi muzsikos lelkek...
(Verebély Pálné)
- 29 Gépfrók a mennyben
(Lampert Csilla)
- 30 „Komplexitás” (Záruba Károly)

SOLARSOFT LEMEZKAU

- 30 SZOFTVERTÉKA
(Szerkeszti és írja: Herczeg József)

- 31 Tárgyszerű víziók
- 31 Szabvány teremője
- 32 A CorelDRAW méltó vetélytársa
- 34 Játéktér: a jó öreg anyaföld
- 36 A Maxis Software sokat adott

KIRAKAT

- 39 Átlépés egy másik dimenzióba
(Sík Zoltán)



MŰHELY

- 42 Organikus, permanens folyamat
(Kiss János—Kereszturi János)

KÖZELGÉP

- 45 A hardver(es) jövője

PROGRAMOZÁSTECHNIKA

- 46 Taszkkommunikációs eszközök
(Dobi Sándor)
- 48 Olvasás innen-onnan, így és amúgy
(Nemes Mihály)
- 49 Ne kezdjük újra! (Vargha Dénes)

KALEIDOSZKÓP

- 52 Számkigyó-bűvölés
(Vargha Dénes)

PRO DOMO

- 54 Volt egyszer egy PC Turbo Klub
(Varga János)

55 BÖNGÉSZDE

56 MIKROBAZÁR

56 KÖNYVESPOLC

PALETTA

- 58 Intelligencia minden szinten
(Sziebig Andrea)

MÁGNESLEMEZ MELLÉKLET

Feleki Zoltán karikatúrái

Címlapképünk
a Grundig AG prospektusából

- 33 E számunk hirdetői

Az egész szakma szerszáma

„Magzarul beszelünk” — ezt a címet adtuk a hónap kiemelt témájának több mint két éve, amikor útjára indítottuk az Alaplapot, s azóta is többször visszanyúltunk az akkor felvetett kérdéshez: mennyire képes a számítógép arra, hogy a magyar nyelvű szövegeket korrektül feldolgozza. S noha ezen a lerághatatlan csonton most is sokat tudnánk még szortyogni, ezúttal a dolog másik aspektusára keressük a választ: mennyire ügyes szerszám mindennapos munkaeszközünk, a magyar számítástechnikai szaknyelv.

Számunkra ez a téma a szokásosnál szubjektívebben vetődik fel, mert egész lapkészítési munkánk alapvető koncepcióját érinti. Szó sincs persze arról, hogy ne adnánk ebben is ugyanúgy teret minden ellentétes véleménynek, de a felvetett kérdésekben az Alaplap a számítástechnikai ismeretek terjesztéséhez alkalmazott stílusával, szóhasználatával, írásmódjával eleve kifejez bizonyos nyelvi koncepciót, tudatos törekvést. Felfogásunkat az olvasó a lap egészéből is leszűrheti, de az egyértelműség — és az esetleges vita — kedvéért ebben a bevezetőben is megfogalmazzuk.

Nyelvi eszköztárunk kialakításához mindenekelőtt végig kellett gondolnunk, hogy várhatóan milyen formában honosodik meg a számítástechnika a magyar nyelvben.

1) Elszigetelt szakmai tolvajnyelv lesz, amelynek nincs is szüksége a köznyelv irányába építendő hidakra? (Lásd vegyészet.)

2) Misztifikált tudományos tolvajnyelv köntösét ölti magára, amely mellett párhuzamosan létezik azonban egy hétköznapi szóhasználat is? (Lásd orvostudomány.)

3) Közérthető szakmai kifejezésekkel dolgozik, s így könnyebben érteti meg széles körben önmagát? (Lásd történettudomány.)

Az első változat inkább csak olyankor életképes, ha egy adott szakterület megmaradhat kevesek mesterségének (harangöntés), eredményeinek élvezetéhez (harangszó) nem szükséges ismerni a gyártási technológiát, de még a késztermék működtetésének módszereivel is csak egy igen szűk rétegnek kell tisztában lennie (harangozó). Talán nem szükséges bővebben fejtegetnünk, hogy a számítástechnika számára ez az út miért nem lenne járható.

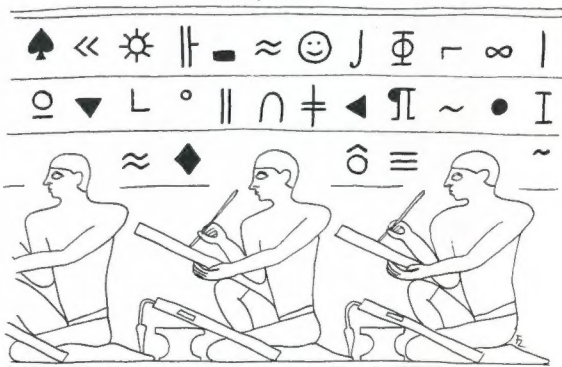
A második lehetőséggel a számítástechnikai szakemberek is sokáig kacérkodtak. A PC megjelenése előtt a nagygépes adatfeldolgozási bémunkát ugyanolyan mesterséges nyelvi kód lebegte körül, mint az orvosi zárójelentéseket. Ez a „kettős könyvelés” a számítástechnikában sokkal nehezebb lenne, mert a torokgyulladásról még csak van fogalmunk annak latin neve nélkül is, de a lemezen lévő hibás cluster megtalálásához nincs semmilyen velünk született „közvetlen laikus eljárás”.

A számítástechnikai eszközök és programok használatához a jelek szerint elengedhetetlen feltétel működési módjuknak — alkalmazási területtől függő szinten való — megértése, s a kettő között a szaknyelv az összekötő kapocs, amit egy eljövendő „fekete doboz” korszak sem tehet feleslegessé.

Szinte kizárásos alapon jutottunk el tehát oda, hogy a számítástechnika szaknyelvének szükségszerű útja a közérthetőség. Ebben nagyon hasonlít például a gazdasági terminológiához, amelyet szintén az élet minden területén kell érteni és használni, mégpedig egységesen, egyértelműen és szabatosan, akár egy nagyvállalati főkönyvelő ír mérlegbeszámolót, akár egy magánkereskedő nyújt be hitelkérelmet.

Miután tehát megfélélsünk szerint a majdan kiforrottá váló magyar számítástechnikai szaknyelvre nem az elkülönülés, hanem a köznyelvhez való szerves illeszkedés lesz jellemző, mi kezdetül fogva igyekeztünk még a számítástechnikai boncnyelven megírt, agyonkódolt szakanyagokból is minél több ember számára érthető, élvezetes írásokat szerkeszteni.

Azzal is tisztában voltunk viszont, hogy a szaknyelvnek is megvan a maga természetes fejlődési folyamata, amit nem lehet erőszakosan és radikálisan megváltoztatni, legfeljebb csak finom lépésekben befolyásolni és korrigálni. Ezért egyaránt elzárkóztunk az erőltetett, mindenáron való magyarítási törekvésektől és az angol terminológiát tűzőn-vízen át előnyben részesítő xenomániától.



Az én nyelvem, a te nyelved, az ő nyelve

A tűzcsapból is víz folyik

A szociológusok szerint a szakember abban különbözik a hétköznapi embertől, hogy időnként olyasmit mond, amit a kívülállók nem értenek meg.

Mi, számítógépesek, jó szakemberek vagyunk, és néha túlteljesítjük a normát: ilyenkor olyanokat mondunk, hogy magunk sem értjük.

Mitől lesz érthetetlen egy mondat? Ennek alapjában véve két oka lehet: vagy ismeretlen szavakat használunk, vagy nem úgy rakjuk össze mondatát a szavakat, ahogy azt a magyar nyelv diktálná. A két jelenségnek egészen más súlya van. Az első, hogy a magyar mondatba idegen szavakat keverünk, egészen mindennapos. Az, hogy a mondat szerkezete nem eléggé magyar, már ritkábban fordul elő, akkor is inkább a fordításokban.

A szótár

A szaknyelv egyik jellegzetessége, hogy szüntelenül új fogalmak tűnnek fel benne. Hogyan hívjuk őket? Az angolok nagy előszeretettel használnak olyan rövidítéseket, amelyek önmagukban is értelmes, sőt az adott fogalmat bizonyos mértékig jól kifejező szavak. Amikor ez nem megy, akkor valami mást találnak ki, de sokkal kevesebbet morfondíroznak rajta. A bit, a chip, a BASIC rendkívül találó szavak. Hogy a stack (rakás/halom) miért az egyik fajta memóriakioldozás neve, míg a heap (halom/rakás) a másik fajtáé, az nyilván csak véletlen műve, hiszen az angol értelmező szótár is az egyikkel magyarázza a másikat és a másikkal az egyiket, számítástechnikailag pedig két elég különböző dologról van szó.

A fogalommal történt megismerkedés után az első lépés nyilván az, hogy úgy használjuk, ahogy leírva látjuk. Itt már kezdődnek a problémák: vagy tudjuk ugyanis a szó kiejtését, vagy magunk „generáljuk”, elképzeljük valahogy. Kérdés az is, hogy amit ismerünk, azt angolul tudjuk vagy amerikaiul. Továbbá hogy milyen végződéseket diktál hozzá a fülnk. Ezen nemigen hiszem, hogy lehetne segíteni. Ami az egyik embernek 'standard', az a másik-

nak 'sztenderd', a scheduler az egyik embernek 'sedjuler', a másiknak 'szkedjuler' — aszerint, hogy korábban hol találkozott a szóval.

Ezután több dolog történhet. Lehet, hogy az idegen szó lassan magyar szóvá válik, valamilyen formában rögzül a helyesírásra, a kiejtésre, a ragozásra, és végül az Akadémia is áldását adja rá. Másik megoldás, hogy a mechanikusan lefordított idegen kifejezést használjuk, de az is előfordul, hogy valaki talál a fogalomra egy jó magyar szót, és az terjed el.

Gyakoriságát tekintve valószínűleg az első út a tipikus, bár minél távolabb van a szó eredeti alakja a kiejtésétől és a magyaros hangzástól, annál nehezebben megy a dolog. A 'mini'-ből pillanatokat alatt magyar szó lett. A makróhoz, lézerhez, processzorhoz sem kellett sok idő. Nehezebben ment mindez a hardverrel, szoftverrel, szervizzel, a fájl szó pedig még ma is éles vitákat vált ki.

Én úgy látom, hogy a sikeres magyartításnak két feltétele van: az egyik, hogy idejében kell megszületnie (mielőtt az idegen szó túlságosan elterjedne), a másik pedig az, hogy az eredeti angol szó ne legyen túlságosan találó, viszont

a magyar szó legyen az. Erre a legjobb példa az 'értékkadás', ami tényleg sokkal jobban megragadja a dolog lényegét, mint az angol assignement (hozzárendelés, kijelölés). A magyar verem szó is sokkal jobb egy olyan tárolóra, amelynek csak a tetejéről lehet elvenni, és oda lehet tenni, mint a stack (azaz kazal), amiből bizony ki lehet húzítani egy kis szalmát oldalról is.

Távol álljon tőlem a nyelvvédők bírálata, de az elmúlt 35 évben megéltém egy tucat magyartítási kampányt, és azokból nem emlékszem egyetlen szóra sem, amely végül is kiszorította volna a kiszorítandó szót.

A fordítás szerintem akkor válik be, amikor az eredeti szónak nincs semmi rejtett jelentése, és a lefordított szó is ugyanazt jelenti. Világos például, hogy a 'line printer'-t nyomtatónak kell nevezni. Persze a korrekt fordítás igen kemény dió. Az elmúlt hónapban két kolléga is hosszú előadást tartott nekem arról, hogy mekkora marhaság a RAM-ot (random access memory) véletlen hozzáférésszerű tárnak nevezni. Egyesek szerint jobb lenne, ha tetszőlegesen vagy találmányok elreható tárnak hívnánk. Hát én ezt sem érzem sokkal jobbnak, hiszen valójában arról van szó, hogy a tár bármelyik egysége közvetlenül, a cím alapján elérhető, így tetszőlegesen, ha akarom véletlenszerűen helyezhetők el a tárn az adatok, és ez nem befolyásolja elérési idejüket. Mindez bizony az eredeti angol kifejezésből sem igen tűnik ki, miért lenne hát a tükörfordítás sokkal jobb. De végül is ez nem tragédia, az elnevezés csak elnevezés, a



— ...az új ansziszisszel sikerült végre bútolni, és rögtön debaggoltam a baszfájll...

tűzcsapból is meg a vízcspapból is víz szokott folyni.

Vannak azután felemás megoldások, mint az operációs rendszer. Vajon miért nem hívjuk kezelő vagy működtető rendszernek? Hiszen az a funkciója. Amennyire vissza tudok emlékezni a dologra, ez az alternatíva már a kezdet kezdetén felmerült, de a többség nem találta elég szakszerűnek. Azt mondták, hogy az orvos is latinul szokta megnevezni az ember szerveit, pedig a legtöbbnek van becsületes magyar neve.

Sajátosan magyar — és elsősorban nyelvészek által javasolt — megoldás a visszatérítés. Vagyis ha az angol szó eleve latin eredetű, térjünk át az angol kiejtésről a magyar hagyományok szerinti latinos kiejtéshez, szóalakhoz. Az egyik ilyen kampány azért indult, hogy ne 'kompjütör' mondjunk, hanem 'komputer'. Anyit értek csak el, hogy sikerült a kiejtés sokféleségét megnövelni. Mostanában többen elmagyarázták nekem, hogy 'projekt' vagy 'prodzsekt' helyett helyesebb volna a 'projektum' használata, hisz ez az eredeti latin szó. Valószínűleg ezzel is csak az előforduló variációk számát növelnénk.

A nyelvtan

Az emberek általában úgy vélik, hogy a szaknyelv sokkal pontosabb, egyértelműbb, mint például az irodalom nyelve. Korábban voltak olyan feltételezések, hogy a szaknyelv egzaktágának forrása a korlátozottabb — vagy legalábbis a hétköznapi beszédétől és az irodalomtól arányában eltérő — nyelvtan használata. Noha pontos statisztikát nem csináltunk, elég sok különböző szöveget elemeztünk géppel, és nem tűnt fel, hogy valamilyen nyelvtani szerkezet hiányozna vagy éppen a szokásosnál sokkal gyakrabban fordulna elő a számítógépes szövegekben. Apreszjant professor, aki rengeteg elektronikai szöveget géppel fordított angolból és franciából oroszra, szintén azt mondta, hogy ha egy nyelvtani konstrukció létezik, akkor azt használják, bármilyen témáról legyen is szó.

Ezzel kapcsolatban szeretném felhívni a figyelmet két nyelvi babonára. Sokan tudják úgy, hogy a szenvedő mondat szerkezet magyartalan, illetve germanizmus. Ez így nem igaz! Ha azt mondom, hogy: „bele van gyűrődve a nyomtatóba a papír”, az ugyanolyan jó, mint az, hogy „zöldre van a rácsos kapu festve” vagy „be van fejezve a nagy mű, igen”. Mindazonáltal ez a szerkezet mégis viszonylag ritkán szerepel az



—Ha így ejtem ki a kompjütört, a nyelvészek talán nem kötnek bele.

alany állapotának jellemzésére. És itt jön a másik babona. Egyesek azt hiszik, hogy magyarosabb lesz a mondat, ha névszói szerkezetűvé alakítják, amiben egy múlt idejű melléknévi igenév az állítmány. Például ahelyett, hogy „az A és B pont a piros vezetékkel van összekötve”, azt mondják, hogy az „az A és B pont piros vezetékkel összekötött”. Hát ez egy óriási marhaság, mert a múlt idejű melléknévi igenévi állítmány igen kivételesen előforduló forma. Számos nyelvész szerint ilyen mondat nincs is, mert múlt idejű melléknévi igenév nem lehet állítmány, csak ha így jellegét elveszítve már melléknévvé vált.

A stílus

A egzaktág mellett a szaknyelvről azt is feltételezik, hogy meglehetősen száraz. A költemények tele vannak érdekes hasonlatokkal, allegóriákkal, megszemélyesítésekkel. A prózai művekben is vannak ilyenek, de elég ritkán fordulnak elő a számítástechnika nyelvén. Mi is meglepődünk, amikor egy szigorúan technikai szövegben először találunk megszemélyesítést. Ha jól emlékszem, egy lemez „haldoklásáról” volt szó, annak kifejezésére, hogy valamely hiba miatt a lemezen egyre több szektor vált használhatatlanná.

Vannak persze szövegek, például a referencia-kézikönyvek, amelyeket azért veszünk kézbe, hogy megnézzük: ezt ide kösd, amazt így állítsd be, emennek ilyen paraméterei vannak. Ilyen esetekben engem is zavarna minden „lazaság”. Legyen az ilyen szöveg teljesen pontos és tömör. Más a helyzet akkor, amikor valamit el kell magyarázni. Azt szokták mondani, hogy egy jó ábra többet ér, mint két oldal szöveg. De a szakemberek közül kevesen tudják, hogy egy szóképp is sokat segíthet valaminek a megértésében.

Egyes sikeres szerzők részletesen kifejezett allegóriákat alkalmaznak egy-

egy bonyolultabb téma ismertetésekor. Különösen szép példákat találunk a multitasking rendszerek leírásánál. A 'task'-ok születnek és meghalnak, gyermekeket szülnek vagy osztódnak, randevúznak egymással, versengenek az erőforrásokért, amelyik pedig nem jut hozzá, azt kiéhezettik vagy kilövik.

Persze felmerül a kérdés, hogy vajon a „kiéhezettetés” hasonlat-e vagy szak kifejezés. Biztos vagyok benne, hogy először az „intelligens” szót is szimbolikus jelzőnek szánták, ma pedig az „intelligens terminál” meglehetősen pontosan, majdhogynem formálisan definiálható. A „barátságos” (felhasználóbarát) szó talán még nem jutott el erre a fokra, de azért már többé-kevésbé pontosan tudjuk, hogy mit jelent.

Összegzés

Mint minden nyelv, a magyar is igen sokféle. Más az én nyelvem, a te nyelved, az ő nyelve, ha én számítástechnikus vagyok, te körbontnok, ő meg alkotmánygőz. Persze egy negyedik témában (különösen, ha egyikünk sem ért hozzá igazán) igen jól el tudunk beszélgetni. És más volt a nyelv tegnap, és más lesz holnap. Még inkább igaz ez a számítástechnika nyelvére, amely a szakmával együtt rendkívül dinamikusan fejlődik. De azért valahol a mélyben minden szaknyelv a hétköznapi magyar nyelvvel azonos szabályok szerint építkezik. Az nem várható el, hogy valaki egy komoly szakcikket minden előzetes ismeret nélkül megértse, már a nyelv szintjén sem, hiszen számára idegen szavakkal és friss magyarázatokkal kellene megküzdnie. De annak, ami ma szaktudás, egy része holnap közkinccs lesz. Ezért mindig keresni kell az új fogalmak szabatos és egyértelmű magyar megnevezését, óvatosan kell bánni a túlkörnyordításokkal, és törekedni kell a szemléletességre.

Farkas Ernő

6:3 helytlen döntetlen

Angol-magyar mérkőzés

Az utolsó egy-két évtizedben a tudományos-műszaki életben talán a számítástechnika fejlődése volt a legsikeresebb és a leglátványosabb. Az új tudomány és annak gyakorlati alkalmazása elkerülhetetlenül felvetette a vele kapcsolatos új nyelvi kérdéseket is.

A 80-as évek közepén a magyar sajtóban csaknem fél évig vitatkoztak arról, hogy milyen, illetve milyen legyen a számítógépes szaknyelv. Csak néhány kiragadott idézet a vélemények közül:

„Itt az ideje, hogy ne rontsuk tovább nyelvünket... már régóta forr bennem a nyelvérzéki indulat két, írva, nyomtatva, olvasva, kimondva képtelen szó ellen... Természetesen a *hardver* és a *szoftver* szóösszetételre gondolok. Míg az eredeti angol *hardware* és *software* alakjukban megjárta, legalább olvasva nem bosszantotta az embert. De fonetikusra átmagyarártva elviselhetetlen is, értelmetlen is.” (Boldizsár Iván: Kemény és lágy. Népszabadság, 1985. XII. 24.)

„Az egységes számítástechnikai szaknyelv... alapja Magyarországon is csak az angol lehet... *Ne ott magyarítsunk*, ahol az erőszakolt magyar szóhasználat az egyes szakterületeken való elmaradottságunk megőrződéséhez vagy fokozódásához is hozzájárulhat. Számoljunk le végre azzal az illúzióval... hogy honi magyar nyelvünk valaha is alkalmas lesz a világ méretű haladás naprakész követésére...” (Oláh Zoltán: Számítástechnika és leányvállalat. Magyar Nemzet, 1985. XII. 29.)

„A nemzetközi hordaléknyelv divatszavainak, a kereskedelem, a vendéglátóipar és a számítástechnika főként angol szakkifejezéseinek beözönlése elleni harc nálunk többintézkedés akadályokba ütközik... Vannak, akik egyenesen nyelvünk gazdagodásának minősítik a minél több internacionalista kifejezés meghonosítását. Ez az álláspont a legveszélyesebb, mert a magyarságtudat elhalványodásához, a nemzeti nihilizmus terjesztéséhez vezet.” (Antalfi Gyula: A számítógép nyelve. Magyar Nemzet, 1986. I. 11.)

„Mindaz, ami a köznyelvben kérhet jogot arra, hogy a számítástechnikai használatnál *analóg* kifejezésként szerepeljen: legyen magyar. Ugyanakkor nem tiltható meg a szaktudománynak, hogy kihasználja a számítástechnikai szaknyelv nemzetközi előnyeit... mert nem a magyart rontja, s mert nem az angol honi szálláscsinálója.” (Ungvári Tamás: Eretek nézetek a szakmai nyelvről. Magyar Nemzet, 1986. II. 18.)

„Az *angolos* terminológia mellett legerősebbnek tűnő érv az, hogy az angol biztosítja a szaknyelv nemzetköziségét. A nemzetközi terminológia azonban mítosz... Nem elég a rossz hangzású szó miatti felháborodás és a javító szándék. Szakmailag meg kell tudni ítélni, hogy a létrehozott szó alkalmas-e az adott fogalom kifejezésére, és van-e lehetőség az elterjesztésére. Ehhez viszont szakértelem kell, nem elég a jó szándék.” (Kiss Ádám: Szoftver és türelem. Magyar Nemzet, 1986. III. 17.)

„Feltűnően sokat beszélünk a számítástechnika nyelvről; hajlok arra, hogy sok esetben azért borzongunk a nyelv szörnyűségétől, mert valójában az új technika birtokbavételétől ódzkodunk.” (Mérő László: A tenni kéne valamit szindróma. Magyar Nemzet, 1986. III. 24.)

A számítástechnika nyelvzetéből írt cikkeknél természetesen csak csekély rész hánnyadából idéztem, de talán ennyi is elegendő ahhoz, hogy nyilvánvalóvá váljék: az egyes vélemények között kibékíthetetlen ellentétek vannak.

A szakemberek jelentős része úgy véli, hogy mindenképpen ragaszkodnunk kell az eredeti angol nyelvű terminológiához, s ennek megfelelően át kell vinnünk és használnunk az idegen nyelvű szakkifejezéseket. Érveik közül

a leggyakoribb a szaknyelv nemzetközisége, s írásaikból az hallik ki, hogy az új technika befogadásához a nyelvi elemek átvétele is szükséges, különben szakmai elmaradottságunk is növekedhet.

Mások viszont élesen kikelnek „a nemzetközi hordaléknyelv divatszavai” és az „angol kifejezések tébolyító dömpingje” ellen.

A dolog szépséghibája mindössze az, hogy sokszor olyan emberek is elítélően nyilatkoznak a számítástechnikai szaknyelvről, akik a *hardver* és a *szoftver* szón kívül egyáltalán nem ismerik a számítástechnikai szakszavakat, számítástechnikai ismeretekkel pedig egyáltalán nem rendelkeznek. Az önmagában helyes, hogy lehetőleg magyar szavakat használjunk, de a szakterület valamely mértékű ismerete nélkül nyilatkozni a szaknyelvről meglehetősen vitatható dolog.

Kissá sarkítva: a hozzászólók egyik csoportja szerint a számítástechnika szaknyelve csak az angol lehet, hiszen illúzió, hogy a magyar nyelv alkalmas lesz e tudomány művelésére. Ellenzők viszont követeli, hogy a szaknyelv is legyen magyar, mert a nemzetközi szaknyelv, a nemzetközileg egységes terminológia nem több, mint pusztá mítosz.

Valójában az eltérő vélemények sokkal árnyaltabbak. A véleményüket nyilvánított minden esetben más-más határt húznának meg az eredeti angol kifejezések megtartása vagy magyarosítása kérdésében. Érdekes viszont, hogy az egymástól homlokegyenest eltérő vélemények egy kérdésben többnyire azonosak: a többség ellenzi az eredeti angol szakkifejezések fonetikus átirását, bár rendszerint eltérő okokból.

A fonetikus átirás kompromisszumként született. Az eredeti angol szavak leírása és helyesírási szabályaink szerinti toldalékolása, azaz a magyar mondatokba való beillesztése ugyanis számos nehézséget okozott (*hardware*-rel, *byte*-ot, *file*-lál stb.). Akadt kezembe olyan számítástechnikai szakkifejezés is, amelyben egy bekezdésen belül egy angol szónak ugyanazon magyar toldalékkal ellátott alakja három formában is szerepelt: *file*-vel, *file*-lál, *file*-le!

A magyartást ellenzők egy része azért tiltakozik a fonetikusán átvett formák ellen, mert szerintük elvesztjük az eredeti angol nyelvi forma használatából fakadó előnyöket, s különben sem a magyar köznyelvben, hanem csak a számítástechnika tudományos nyelvben használják őket.

A magyartást kívánók többsége szerint viszont éppen az a baj, hogy jöllehet elvesztettük az eredeti angol formák ismeretét, mégsem magyarosítottuk meg a szóba jövő kifejezéseket, hiszen a *hardver* vagy a *szoftver* — és sorolhatnám még a példákat — a bennük rejlő mássalhangzó-torlódás és a magánhangzók diszharmonizációja miatt, pusztán az írásképet megváltoztatva, semmivel sem lett magyarabb, tehát semmivel sem kerültünk közelebb a problémák megoldásához.

Mindkét táborban voltak és vannak olyanok, akik elfogadják ezt a — mondjuk ki nyugodtan — félmegoldást. Ezt bizonyítja az MSZ 7788 jelzetű, „Az adatfeldolgozás fogalmai” című, a számítástechnika különböző részterületeinek terminológiáját szabályozó szabványosorozat, amelyben szép számmal találhatók fonetikusán átvett szavak.

Véleményem szerint az egyes szavak fonetikus átírása nem oldja a nehézségeket, az ellentéteket, hanem tovább bonyolítja a helyzetet. A szakíróknak, a szaknyelv használóknak nem csak az eredeti angol írásképet kell ismernie, hanem azt is fejben kell tartania, hogy az adott szakkifejezést csak angolosan, csak magyarosan, vagy esetenként mindkét formájában leírhatja-e.

A fonetikus átírás ilyen veszélyeket hordozó mivolta más szakterületen már bebizonyosodott. A Brenscsán János által szerkesztett *Új orvosi szótár*-ban látunk erre jó — azaz rossz — példát: 63 *pseudo* előtagú szóösszetétel közül 36 esetében utalja az olvasót a *pseudo* előtagú címszóhoz. Ez azt jelenti, hogy 63 eredetileg azonos előtagú szóösszetétel közül 27 szakszóban — a *pseudo*, 36-ban pedig a *pseudo* írásmódot kívánja meg. Kétve hiszem, hogy bárki is fejben tartaná, melyik összetett szó előtagjában melyik írásképet tartja helyesnek a szótár. S kőve hiszem, hogy a számítástechnikai szaknyelv amúgy is meglehetősen sok tisztázatlan nyelvháztalálattal kezdődött hasonló írásképi szavakkal kell még tovább bonyolítani.

A számítástechnika szaknyelve — más szakterületek nyelvétől eltérően — sajátos kettősségeket hordoz. Maga a szaknyelv is kettős: a programnyelvek (gépi nyelvek) angol eredetűek, az angol írásmókat őrzik, míg a

tágabb szaknyelv is angol eredetű, de számos magyar elem is található benne. A programnyelvek magyartásáról érdekes módon nem esik szó. Ehhez ugyanis nem nyelvi, hanem számítástechnikai fejlesztés, azaz önálló magyar számítógépgyártó ipar kialakítására lenne szükség. Eltekintve a gazdasági kérdésektől, erre csak akkor lenne érdemes vállalkozni, ha a magyar termék színvonal jelentősen felülmúlja a jelenlegi világszínvonalat. Erről pedig jelenleg ábrándozni sem érdemes.

A másik kettősség, hogy az angol szavakat a magyar közleményekben magyar szövegekbe, mondatokba helyezzük bele, s ennek megfelelően magyar toldalékokkal (ragokkal, jelekkel, sőt: képzőkkel is) látjuk el őket.

Nem az a baj igazán, hogy idegen szavakat veszünk át, hanem az, hogy ezek nem illeszthetők jól be a magyar kiejtési és grammatikai rendszerbe.

További ellentmondás, hogy egy-egy angol szót (*cursor*) a magyarok eléggé sokféleképpen ejtenek (kurzor, kőrsz, stb.), sőt időnként az írásmóval is gondok vannak. (Pl.: a szövegi néma magánhangzóhoz kötőjellel kapcsoljuk a magyar toldalékokat: *hardware*-es!)

A harmadik, hogy a számítástechnikai szaknyelvet használók köre két részre oszlik. Az egyik, a számítástechnikai szakemberek, akik otthon vannak saját tudományukban, s ennek megfelelően kimerítően kell ismerniük szakterületük minden nyelvi jelenségét. Legyünk őszinték, s mondjuk ki, hogy megfelelő angol nyelvi ismeretek nélkül igazán ez senkinek sem sikerülhet.

A másik kör a felhasználók köre. Hovatovább a számítástechnika minden tudományba betör, s az összes tudományok összes művelője, ha érdemben hozzá akar járulni szaktudománya fejlődéséhez, kutatómunkájában nem nélkülözheti a számítógép segítségét.

De nemcsak a tudományban, hanem a mindennapi gyakorlati munkaterületeken is jelentős segítséget, könnyítést nyújt a számítógép használata a kereskedelemről a gyógyításig, az oktatástól a bűnüldözésig.

Ezeket a felhasználókat nem igazán érdekli vagy izgatja, hogy a számítástechnikai szaknyelv valójában milyen. Számukra nem érdekesek az erről folytatott viták sem. Nekik arra van szükségük, hogy kutató- vagy gyakorlati munkájukban hathatós segítséget kapjanak a számítástechnikától. Azaz olyan programok álljanak rendelkezésükre, amelyek megkönnyítik munkájukat. Ha ehhez kevesebb számítástechnikai

szakismeretre van szükségük, szívesebben és könnyebben használják a számítógépet. Ha a munkájukhoz szükséges számítástechnikai ismeretek megszerzését bonyolult nyelvi ismeretek elsajátítása vagy egyszerűen a terminológiai zavarok meglete nehezíti, akkor egyértelmű, hogy bizonyos ellenállás mutatkozhat a számítástechnika használatba vételével szemben.

Más mércével kell tehát mérnünk, attól függően, hogy melyik körnek szól a közleményünk. A számítástechnikai szakemberek körén belül szinte mindig, hogy milyen nyelven folyik egy vita, csak a kölcsönös megérthetőség és az egyértelműség követelményének kell minden esetben eleget tenni.

Ezen túl az, hogy egy-egy műhelyben, pl. egy szaktanulmányban a terminológiai választékból melyik kifejezést használják, az az ott dolgozóknak szinte magánügye. Viszont egy publikáció esetén az érthetőség miatt már valamilyen szaknyelvi normához kellene tartanunk magunkat, s ez nem mindig sikerül.

Közbevetőleg jegyzem meg, hogy a nyelvi norma megsértése nemcsak a terminológia vagy a helyesírás területén zavaró. Talán itt a legkevésbé, hiszen még nem is beszélhetünk egyértelműen kodifikált számítástechnikai terminológiai és helyesírási normáról. Számomra az még inkább zavaró, hogy sok szak-közleményben az általános magyar nyelvi normákat sem tartják szem előtt, s a fogalmazási, nyelvi-nyelvtani, szóhasználati pontatlanság, sőt időnként a helyesírási hiba is.

Mi tehát a megoldás? Az ideális a svéd példa követése lenne. Jól felkészült szakemberek (nyelvészek is!) foglalkoznak egy kutatóközpontban azzal, hogy az újonnan megjelent, idegen nyelvű szakszavak, szakkifejezések helyett megfelelő, a nyelvi követelményeket kielégítő svéd nyelvi elemeket készítsenek és segítséget nyújtsanak azok elterjesztésében, illetőleg a normának nem megfelelő nyelvi alakzatok kiszűrésében.

Ez az út nálunk egyelőre nem járható. Kár, mert minden szaknyelv fejlődésében nagy segítséget nyújthatna egy hasonló magyarországi intézmény felállítása. Ennek azonban nemcsak személyi, hanem anyagi feltételeit sem tudjuk megteremteni a közeljövőben.

A másik út, amelyen eddig is jártunk — több-kevesebb sikerrel —, az, hogy maguk a szakterületet művelő szakemberek több figyelmet fordítsanak a szaknyelv normatív és terminológiai kérdéseire. Ha ez pusztán annyit jelent, hogy

az egyes szakközlemények megfogalmazásakor igyekeznek gondosabban ügyelni a nyelvi megformálásra, már jelentős lehet az előrelépés.

Alapvető kérdés, hogy angol vagy magyar legyen a terminológia. Nem tartom túl szerencsésnek ezt a merev kettévágást. Igenis átvehetjük és megtarthatjuk az eredeti angol formát, ha az nem okoz különösebb kiejtési vagy írásbeli nehézségeket (*bit*).

Próbálhatunk magyar megfelelőit alkotni abban az esetben, ha az angol frászkép és kiejtés a magyar anyanyelvű számára teljesen idegen, és a szó magyar használata bizonyos zavarokkal járhat (*file-lal, fájljal!*) Ez azonban nem sikerült és nem is fog sikerülni minden esetben. Hiszen egy bizottság, amely szakemberekből és nyelvészekből állt, több napon keresztül vitatkozott azon, hogy a *hardware* és *software* angol szavak helyett mit ajánljon. Végül a fonetikus átírásnál jobbat nem tudott javasolni, pedig több tucat javaslatot kapott ülései során. S ma e két szó szinte a mindennapi szóhasználat szerves részévé vált, sőt képzett alakjai (*hardveres, szoftveres*) is szerepelnek már szókincsünkben.

A számítástechnikai szaknyelv jelenlegi állapota még a legszigorúbb purista szempontból nézve sem rosszabb, mint sok más szakterület nyelvélé. Jóllehet mindent el kell követnünk, már csak a színvonalas szakmai kommunikáció érdekében is, a szaknyelv fejlesztése, egységesítése érdekében, de helyzetét nem érzem tragikusnak vagy rossznak.

Figyelembe kell venni azt is, hogy szinte a világon a számítástechnika, illetve annak alkalmazása fejlődik talán minden szakterület közül a leggyorsabban. Ez nyelvi szempontból azt is jelentheti, hogy bizonyos, sokak által kifogásolt szavak vagy kifejezések éppúgy a lomtárba kerülnek, mint azok az eszközök vagy eljárások, amelyek jelentését hordozzák.

Arra is számíthatunk, hogy a fejlődés mindig új nyelvi kérdéseket is felvet, vagyis állandó az új angol szakkifejezések és szakszavak utánpótlása. Ha egy új fejlesztés, egy új technikai elem vagy eljárás átvehető figyelembe vesszük — vagy legalábbis megpróbáljuk figyelembe venni — a szaknyelv, tágabb értelemben véve a magyar nyelv fejlődésének a követelményeit, nő az esély arra, hogy belátható időn belül egy jobb, tisztább, esetleg egyszerűbb, magyarosabb, a felhasználók számára is könnyebben érthető szaknyelvet alakíthatunk ki.

Seregly János

A „beépített ember”

Olvass.el (!)

Ugye, milyen sokszor találkozunk a cikk című válaszottott fájlnevvel?

Es ugye, időtlen a külön zárójelbe tett felkiáltójel a végén?

Ha ugyanis fel akarom szólítani valamire az olvasót, akkor miért tompítom mondandómat egy zárójellel,

ha pedig csak mellékesen, zárójelben akarok valamit megjegyezni, akkor miért egy kiugrató jelet teszek bele.

Ez a példa egy a sok közül

a szerkesztés során keletkező nyelvi ellentmondásokra, amelyek feloldása igen gyakran

az olvasószerkesztő feladata.

Az alábbi eszmefuttatás az ő nézőpontjából igyekszik bepillantást adni a szaknyelv gyúrásának küzdelmeibe.

Az olvasószerkesztőt segédszerkesztőnek is szokták nevezni, ami jól utal kettős helyzetére. Egyrészt ő a főszerkesztő jobbkeze a kéziratok gondozásában, másrészt ő az olvasó „beépített embere” a szerkesztőségben. Az egyik gazdasági lap főszerkesztője soha nem korholta a szabadkozó fiatal olvasószerkesztőt, ha az nem értett valamit egy cikkben, hanem inkább dicsérte: „Fiam, ez a dolgod, hogy jelezd! Ha te nem értd, akkor nagy a valószínűsége, hogy az olvasó sem fogja érteni, tehát korrigálni kell a fogalmazást.” Ha csupán nyelvi ügyetlenség zavarja a megértést, az a rutinosabb olvasószerkesztő maga is ki tudja javítani, de ha szakmai értelmetlenségek, logikatlanságok, ellentmondások vannak mögötte, akkor bizony konzultálnia kell a szerzővel.

Szabad a pálya

A fentieket nem azért írtuk le, hogy népszerűsítsünk egy olyan pályát, amely felé nem nagy a törekedés. (Bár az sem ártana, hiszen az Alaplapnál is betöltetlen az olvasószerkesztői státusz, és a szerkesztőknek kell elvégezniük a végső cizolás műveletét, egyik lap-társunk kiadja pedig immár csaknem fél éve kínálgatja hiába az olvasószerkesztői állást a Magyar Sajtó c. lap hasábjain, miközben újságírók száza vannak munka nélkül!) Inkább arra igyekszünk felhívni a figyelmet, hogy a számítástechnika elterjedése lényege-

sen megváltoztatta az írásos dokumentumok, műszaki leírások, kiadványok készítésének menetét. Tömegével kerülnek forgalomba úgy az anyagok, hogy nem is látja azokat hivatásos szerkesztő, olvasószerkesztő, lektor vagy korrektor, ezáltal a legértékesebb tartalmat is agyonütheti a nyelvtani, stílusi és helyesírási hibák tömege. (A bajt még tetézheti a „lézerprinteresített” tipográfiai analfabetizmus... de az már egy másik mise.) A helyzet megváltoztatására sok-sok „amator” olvasószerkesztőnek kellene működnie mindenütt, ahol kéziratokból bármilyen dokumentum érlelődik. Közülük azután néhányan talán ebben lennének meg igazi hivatásukat is.

Egy számítástechnikai szaklap olvasószerkesztőjétől joggal várható el, hogy tudjon magyarul — és természetesen angolul, németül, latinul, görögül... no meg basicul, prologul, pascalul, modulául, adául, lipstul, fortranul, assemblyül, pétegyul, clipperül, cépluszpluszul... De azért mindenekelőtt magyarul. Hasznos, ha volt hozzá egy kis előiskolája is, mert szükséges lesz a helyesírásra, a stílusizikára, a nyelvhelyességre, a szintaxisra, a szemantikára, az etimológiára... És egy jó adag természetes nyelvérzékre.

Egy nyelven beszélni

Ugorjunk csak vissza az iménti nyelvi felsorolásra, hogy mennyi megoldatlan

magyarosítási, helyesírási és kiejtési probléma sűrűsödhet össze néhány szóban is. Sok vitás kérdésben az olvasó-szerkesztő persze nem lehet döntőbíró, sőt a kapott tanácsok is ellentmondóak, de a lapnak mindenképpen meg kell jelennie, és valahogy le kell írni azt is, amiben még sem a gyakorlat, sem a nyelvészek elvi álláspontja nem egyezés.

Az a legjobb irányító, ha mindig mindennek az értelmét, a jelentését keressük, mert a nyelv ezen keresztül szinte automatikusan ellenőrzi a tartalmat. Nincs nagyobb öröm egy olvasó-szerkesztő számára, mint amikor egy olyan szövegben, amelyhez igazán mélyen tulajdonképpen nem ért, pusztán a nyelvi okoskodás alapján felfedezi a szakmai tévedést.

Mindenki közös érdeke, és egyben a távlati cél is mindenképpen az, hogy az egész szakma „egy nyelven” beszéljen. Ez taktikailag valószínűleg úgy közelíthető meg, hogy előbb legalább a nagyobb „műhelyek” (könyv- és lapkiadók, szoftvercégek, oktatási intézmények stb.) igyekezzenek házon belül egységes terminológiát és írásmódot alkalmazni, később pedig sor kerülhet

egy „csúcstalálkozóra”, és annak alapján a normák folyamatos összehangolására.

A szándék szelleme

Természetesen lehetetlen minden cikket a magunk szájze szerint teljesen átírni, de lehet azokat bizonyos nyelvi normákhoz úgy igazítani, hogy megőrizzük a szerző stílusát, a mű eredetiségét. Az út viszont, amely idáig vezet, igencsak hosszú és keserves:

— meg kell érteni (de nem megtanulni) azt a szakterületet, amelyikkel a cikk foglalkozik,

— követni kell az akadémiai iránymutatásokat (amíg csak lehet), de nem szabad visszaradni a saját nyelvi normák kialakításától,

— folyton fel kell tennünk kétkedő kérdéseinket, hogy érthető-e mindaz, ami le van írva,

— végül pedig rá kell szólni az időt a bizonytalan kérdések tisztázására, minél több mérvadó vélemény ütköztetésével.

A hazai számítástechnikai lapok egy részénél a hőskorszakra jellemző lelkesedéssel vezettek egy folyamatosan ak-

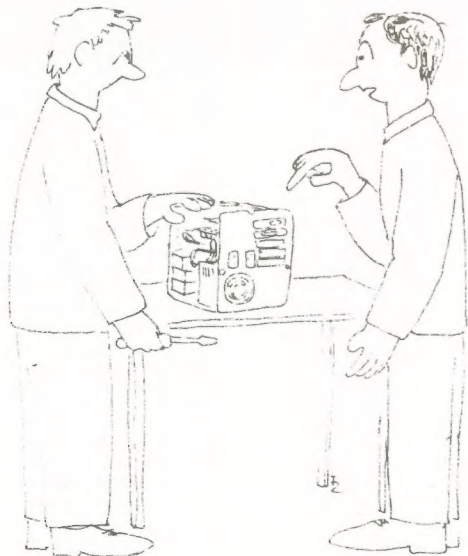
tualizált „szokványjegyzéket”, amiből ki lehetett puskázni a lapba bekerülő új szakkifejezések egységes írásmódját. Ez a „mit-hogyan” gyűjtemény azonban máig sem állt össze igazi számítástechnikai nyelvhelyességi szótárrá, valószínűleg azért, mert egy ilyen vállalkozás a benne részt vevőktől összefogást, a kenyéradó gazdától független együttműködést és elég sok többlet-energiát igényelt volna — s mindez nem volt meg. Talán csak a szándék szelleme örökött azon, hogy ennek ellenére valamennyi szaklapnál megmaradt egy olyan szemlélet, amely a lényegi kérdésekben hasonló elveket tükröz.

Mindenki szavaz

A szaknyelv magyarításában mindig realitásnak kell lennünk. Ha egy új szakmai fogalom bukkán fel, és olyan magyar szót találunk rá, amely pontos, kifejező és jól használható, azt a szakma is nyomban elfogadja, s csak a „pufferben” őrzi meg az angol eredetét. Ehhez viszont valóban titelalálatnak kell lennie...

A „mindent mindenáron magyarul” megoldás gonosz csapda, a szakmai nyelvhasználatról túlzottan idegen magyar kifejezések erőltetése éppen fordított hatást ér el: nevetéssé válik, ami kihat még a későbbi — és esetleg már valóban jó — kísérletek fogadtatására is. A szakmai nyelv fejlődése öntörvényszerű folyamat, s bár valamennyien részesei vagyunk a törvényalkotásnak, a betérjesztett indítványoktól az egész „parlament” szavaz. (A szakmát pedig nem lehet leváltani, még akkor sem, ha a döntéssel elégedetlenek vagyunk.)

A hazai számítástechnika hajnalán a „bájtűfűtők” körében kialakított bükkfanyelv oka nem annyira a majmolás volt, mint inkább a hiányos tudás. Nem is születhetett meg olyan ütemben e szakterület nyelvi infrastruktúrája, amilyen sebességgel jövővényként behatolt társadalmunkba. Ahogy azonban ilyenkor lenni szokott, igen sok kóklér is szakembernek tűnhetett, ha képes volt a divatos szövegáradék durrogtatására. Ma már, a piac kibővülése és nyitottá válása óta a pusztán nyelvi kódóstitéssel manipulálók korszaka is lejárt. Egyre több mindent lehet egyszerű hétköznapi nyelven, mégis egyértelműen és pontosan megfogalmazni, s egyre kevesebben hiszik azt, hogy a szakszerűség és tudományosság a számítástechnikában kizárja a közérthetőséget.



—A ventilátorig még értem, amit mondtál...

Faklen Pál—Varga János

A keresés kritikus kritériumai

Szövegekben böngészve

Tapasztalatom szerint a nyelvészek, ha valami problémájuk van, először más nyelvészek írásaihoz fordulnak, azután beszélnek róla kollégáikkal, és végül, de nem utolsósorban (sőt talán elsősorban) céduláznak: éveken keresztül, sokszor egy életen át gyűjtik az idézeteket az egyes nyelvészeti kérdésekkel kapcsolatban.

Én viszont, mint számítógépen nevelődött ember, ilyenkor is rögtön „a komputeremhez kapok”.

Mióta a magyar nyelvű interfésszel és a gépi fordítás kérdéseivel foglalkozom, minden gépen, ahol csak dolgozni szoktam, van vagy tíz-húsz megabájtnyi szöveg. Ezeknek azután neki lehet esni szövegszerkesztővel, keresőprogrammal, szöveges adatbázis-kezelővel vagy különböző ad hoc programokkal, és megnézni, hogy mi is van bennük — túl a szövegek tartalmán. A piac már számtalan ilyen kereső-elemző programot kínál a szövegszámlálókhoz, és az igény is egyre nő irántuk. Legtöbbször azonban nyilvánvalóan egészen más céllal keresgelnek a számítógépen tárolt szövegekben, ezért talán érdekes lehet számukra, hogy én mire figyeltem fel.

A kóddal nincs szerencsénk!

Eddig a magyar szövegek 60-65 százaléka CWI kódban volt, 25-30 százalék magyar Ventura kódban, s csak a maradékon osztozkodtak az elszigetelt egyéb kódkiadások. Most a Windows rohamos terjedésével és a 852-es tábla megjelenésével újabb problémáknak nézünk elebe. Itt nem akarom részletezni, miért személt (angolul: garbage) az egész 852-es, aki esetleg még nem tudná, olvassa el az Alaplap 1992. áprilisi számában vagy a magyar Chipben. Az új kód eléggé csetelhet ahhoz, hogy a magyar gyomor bevegye, de ilyen téren másút sem meggyújtató a helyzet. A németek is széles körben használják a német ASCII-t, ezáltal a C programokban umlautos betűk jelennek meg a szövegletes és kapcsos zárójelek helyén.

Mindenesetre egy ilyen témájú és drága pénzen magyarított programnak igazán lehetne cserélhető kódtáblája, azaz lehessen meghatározni, hogy melyik kódszámhoz milyen karakterkép tartozzon. Az sem árt, ha egy programnak van billentyűzetátdefiniáló programja, hogy a magyar betűket is le tudjuk vele írni, de még jobb, ha a megszokottakat használhatjuk.

„Keresni csak pontosan...”

A visszakeresés alapfeladata bizonyos szavak és szókombinációk megtalálása a szövegben. Tegyük fel, hogy egy hírózónából ki akarjuk válogatni az almatermeléssel kapcsolatosakat. Ha csak annyit írunk, hogy keresd az „almatermesztés” szót, valószínűleg igen gyatra eredményt kapnánk. Ennek az az oka, hogy a magyar ragozó nyelv. Az angolban egy főnévnek legfeljebb két alakja van (egyes és többes szám), egy magyar szó viszont — amint azt az Alaplap 1992. májusi számában sokoldalúan bemutatták — több ezer alakot ölthet. Még szerencse, hogy a ragok a szó végén vannak, mert így minden szöveges adatbázisban keresgélhetünk jobbról történő csonkolással. Ha tehát azt írjuk, hogy „almatermesztés”, akkor megtaláljuk az „almatermesztéssel”, „almatermesztését”, „almatermesztési” stb. szavakat is.

Persze így sem találjuk meg az olyan mondatokat, hogy: „Almából az idén kevesebbet termesztettek.” Ez a mondat rögtön féltaut problémát vet fel. Az első az, hogy a magyarban számos kapcsolatot lehet egybe- és különírni.

Egybefűjük a jelöletlen birtokos kapcsolatot (lásd a fenti példát), a jelöletlen határozói kapcsolatot, de különírjuk pl. a birtokos és határozós a jelölt kapcsolatot. Az igeikötő és ige különírása és egybefűrése is ismert. És ez nemcsak az igére vonatkozik, hiszen a „megoldható” egybefűrandó, de az „oldható meg” kettőbe. Ilyenkor azt kell előírni, hogy két szót keresünk, s persze nem akárhol. A magyarban természetes egység a mondat, tehát a mondatokat kell felismerni, megjegyezni.

Kaméleon betűk

A szöveges adatbázisok gyakran rendelkeznek olyan lehetőségekkel, hogy a keresett szó állhat a másik előtt vagy után, álljon ennyi meg ennyi szón belül. Ennek a magyarban igen ritkán van értelme, legfeljebb a többszavas nevek-nél.

A mondatot a legtöbb program úgy tekinti, hogy a szövegnek pont, felkiáltójel vagy kérdőjel által határolt része. Persze egy tizedespont vagy egy rövidítést lezáró pont kissé megzavarhatja a dolgot. Biztosabb az algoritmus, ha azt is ellenőrzi a program, hogy van-e utána szóköz és nagybetű, de még így is akad benne elég sok bizonytalanság. (Például tulajdonnév jön a ponttal lezárt rövidítés után.)

Tehát egy mondaton belüli szópárokat kell keresni, például azt, hogy „alma\$” és „termesztés\$”, de „az almának a termesztését” mégsem fogjuk megtalálni, mert az „alma” nem „almá”. Amikor a magyarban egy szó egy raggal találkozunk, gyakran az egyik vagy a másik, esetleg mindkettő átalakul. Jellemző változások: magánhangzó hosszabbodása (béke — békét), magánhangzó rövidülése (kéz — kezét), magánhangzó kiesése (borjú — borjak), a szöveg teljes megváltozása (hisz — higgyük, szó — szavak) vagy egyidejűleg több ilyen jelenség (szép — szebb) stb. Vannak továbbá olyan esetek, amikor a kiejtés nem változik, de a betűkép igen (hűsz — hússzor). Kérdés, hogy milyen gyakori ez a jelenség, és hány távolítatot kell keresni egy szónál. A szavak jelentős részének egyetlen töve van, egy tucat ígének hat

lehetséges tövével találkozhatunk, de átlagosan 1000 töré 2500 tóvátolatat jut. Persze ha van benne némi gyakorlatlanság, és ha egy kicsit utánagondolunk, meg tudjuk mondani, hogy egy adott esetben milyen tóvátolatokat kell keresnünk, a dolog azonban semmiképpen nem megy automatikusan.

Lépjünk egyet tovább. Akkor sem fogjuk megtalálni az „almatermesztést”, ha „jonatánalma-termesztés”-ről lesz szó. A „jonatánalma” esetében mindkét oldalon történő csonkolásra van szükség, ami kicsit munkáigényesebb, ezért nem minden programban van megoldva.

Helyesírási akadályverseny

Tényként kell elfogadni, hogy a forgalomba kerülő programok túlnyomó hányada angol (pontosabban amerikai angol) vagy német nyelvi közeg számára készült, a magyar nyelv követelményei sok szempontból eltérőek: nálunk bizonyos eszközök hiányozhatnak, mások viszont feleslegesnek tűnnek. Remél-

jük, hogy néhány hónapon belül elkészül egy magyar nyelvű szöveges leképező interfész terve, amit majd kivitelezni is kell, de a programokat addig is hozzá lehet mérni.

Ha azt hisszük, hogy egy kinyomtatott könyv, újság, törvény vagy szabvány helyesírása hibátlan, természetesen tévedünk, de a helyzet ennél is sokkal rosszabb a belső dokumentumok, fordítások, jelentések és hasonló kéziratok esetén. Tulajdonképpen az lenne célszerű, ha visszakeresés előtt minden szöveget végigpásztáznánk egy szóellenőrző programmal, és kijavítanánk. Elég nagy munka, de ha a szöveget sokat forgatjuk, érdemes elvégez-nünk.

Visszakeresés szempontjából két dolog kritikus: az egybe- és különírás, valamint a hosszú és rövid betűk. Az egybe- és különírás helyesírási szabályozása kissé laza, célszerű tehát, ha minden eshetőségre felkészülünk. (Annál is inkább, mert ezeket a hibákat a helyesírás-ellenőrző programok sem tudják kiszűrni.) A hosszú és rövid

betűk megkülönböztetésének esélyét pedig rontja az a dogma, hogy a magyar helyesírás a beszélt nyelvet tükrözi. Ez ugyanis legfeljebb tendenciájában igaz. Kétve hiszem, hogy az „egyett” valaki is képes lenne egy gy-vel kiejteni. Ezenkívül a kiejtés vidékenként, szakmánként és sok más módon ingadozik. Különösen sok baj van az f-vel, például a típus vagy a színes leírás rövid i-vel szinte tőpushiba. Ugyanakkor a hosszú és a rövid magánhangzó a képzett vagy ragozott alakoknál is gyakran váltakozik (tíz — tizedes). Még a jó nyelvérzőknek is gyakran kell a helyesírási szótár után nyúlniuk, hogy elkerüljék a helytelen írásmódot.

Az automatikus szöveg-visszakeresés nincs olyan jól megoldva, ahogy azt a prospektusok nekünk tájálják, de talán nem annyira borzasztó, ahogy a fentiek alapján esetleg azok gondolhatnák, akik még nem használtak ilyen programokat. Én minden fogyatékoságuk ellenére tulajdonképpen elég jól elboldogulok velük.

Farkas Ernő

Aknamezőn lépkedve

Helyesírási gerillaháború

Ezt a cikket csak az **ne** olvassa el, aki eddig még minden számítástechnikai kifejezéssel azonosan írt formában találkozott; akinek soha nincsenek kétségei egy szakszó helyes írásmódjáról; aki megoldottnak tekinti a számítástechnikai szaknyelv helyesírási kérdéseit; és aki az egész kérdéskörnek semmi jelentőséget nem tulajdonít.

A számítástechnikai szaknyelv helyesíráásával kapcsolatban bizonyos határig támpontot adnak a magyar nyelv helyesírása vonatkozó általános szabályok. A gond az, hogy ennek a határnak már az innessé oldalán is van néhány „akna”, a túlsóoldalon pedig elég ingoványos a terep, és nincs hozzá sem térképünk, sem kalauzunk. A hivatalos nyelvészeti dokumentumok szerint ugyanis azokban a kérdésekben, amelyekre vonatko-

zón az általános szabályzat nem ad eligazítást, a szaknyelveknek módjukban áll saját normáikat kialakítani. Ez a szemlélet sajnos minden szakterületről feltételezi, hogy megvannak az egy-egy szaknyelvi platform megteremtéséhez szükséges körülmények. Hát a mi esetünkben ez elég nagy tévedés!

Az írásmódban kialakulhat az egység megegyezései alapon vagy diktált módon. A számítástechnikára azonban sem

az egyik, sem a másik út nem jellemző. Még egyazon intézményen belül sem nagyon lehet egységet teremteni, mert aki egyszer feyvert (programnyelvet, szövegszerkesztőt, kiadványszerkesztőt) kapott a kezébe, az rögtön egyszemélyes gerillacsapatnak képzelte magát, és meg akarja vívni külön kis harcát. Ő maga sem tudja ugyan, hogy ki és mi ellen (hiszen többnyire még a magyar helyesírás általános szabályainak is mellőző, és elszigetelten, egyedül durrogat az erdőben), de a keletkező irományokba pillantva elképesztő helyesírási kép tárul elénk.

Valamivel jobb a helyzet, amikor a számítástechnikai íráskor könyvből, folyóiratból jelennek meg, mert olyankor többnyire belép a folyamatba egy műhely (például szerkesztőség, nyomda), amely az általa kialakított normákat bizonyos fokig érvényesíteni tudja. De a különböző műhelyek közötti már szintén nincs szervezett egyeztetés. Azok is csak egymás ellen elszigetelten harcoló partizánalakulatok — több emberrel és nagyobb kaliberű fegyverekkel.

Elvileg ugyan semmi akadályja annak, hogy a gerillaegységek képviselői egyszerű mégis közös tárgyalóasztalhoz üljenek, de erre jelenleg hiányzik a belső készletés, hiányzik hozzá a közvetítő és szervező munkát vállaló intézmény (ENSZ), és legfőképpen nincsenek meg tételek listába szedve

a megtárgyalandó alapdokumentumok.

Magunk részéről az egyeztetés belső igényének felkeltéséhez legfeljebb lapunk egészével tudunk hozzájárulni, remélhetőleg nem a rossz, hanem a jó példa hatásmechanizmusát működtetve. A szervezés gazdájának megtalálásához e havi kiemelt témánk cikkeivel próbáltunk közelebb kerülni, de az ügy érdekében a legtöbbet talán mégis azzal tehetjük, hogy teret adunk a megtárgyalandó kérdések felvetésének és azok előzetes megvitatásának. Ez utóbbit nyomban el is kezdjük, s igyekszünk majd folytatni is, örömmel fogadva, ha bárki hozzáteszi a maga problémafelvetéseit és javaslatait, vagy vitába száll mások álláspontjával.

Az állománynév és környéke

Kezdjük az egyik leggyakoribb dilemmával, amelyre vonatkozóan a Magyar helyesírás szabályai semmilyen eligazítást nem adnak. Hogyan írjuk le az állományok teljes nevét? Csak annyit bizonyos, hogy a névben lévő pontnak szerepelnie kell, mert az a jelrendszer része, megkülönböztető utalás a fájlnevekre és annak szerkezetét felépítésre. Írásmódja ezen túlmenően azonban már nem egységes. Vegyük sorra a lehetséges megoldásokat:

- autoexec.bat
- AUTOEXEC.BAT
- Autoexec.bat
- Autoexec.Bat
- autoexec.BAT
- AUTOEXEC.bat

Leggyakrabban az első hárommal találkozhatunk, legritkábban az utolsó kettővel. A csupa kisbetűs és a csupa nagybetűs változat azért terjedt el annyira, mert az operációs rendszerek és a fájlkezelő programok csak ezt a két formát alkalmazzák — az egyik így, a másik úgy.

Addig ezzel a kérdéssel nem is kell senkinek foglalkoznia, amíg a parancsok szintjén mozog, mert a programok automatikusan átírják a nevet a saját ízlésük szerint, akárhogy gúpeljük is be azt. Szövegyszerkesztésben viszont már „mindent szabad”, s választás elé kerülünk: köznévként vagy tulajdonnévként kezeljük-e? Hm... Ami az iménti konkrét példát illeti, az bizony közelebb van a köznévhöz, mert minden számítógépben ott van, magunk alakítjuk ki, nem fűződik hozzá szerzői jogok, s tulajdonképpen köznévi értelemben használjuk: „Beírtam az elérési

útvonalat az autoexec.bat-ba.” Mintha csak azt mondanánk, hogy „új hengerfejet tettem a motorba”.

Ezzel szemben az állományok másik (jóval nagyobb) csoportjába tartozók egyediek, szerzői joggal védettek, s az írásmódtól függetlenül inkább tulajdonnévi értelemben használatosak: „A program a *kedit.exe* állománnyal indítható el.” Ez a kis kezdőbetű ellenére inkább azzal az autós mondattal analóg, hogy: „A hegyi útrához érdemes *Pirelli* gumit felrakni.” Annál is inkább, mert a *Kedit*, mint az egész programcsomag elnevezése szabályos tulajdonnév.

Az persze teljes káoszt okozna, ha egyenként kellene eldönteni a fájljokról azok nyelvi természetét. (Talán nem is mindig lehet!) Akkor pedig mi a megoldás? Egy kalap alá kell venni minden fájlnevet. Igen, de melyik változat szerint?

1. Az **autoexec.bat** változat azzal jár, hogy minden fájlnevet köznévként tekintünk, ami természetesen nem szerencsés, mert az állományok többsége mégis inkább tulajdonnév. Felsorolásokban ez nem okoz zavart, de mondatban elhelyezve ellentmondásos, nehezítheti a megértést.

2. Az **AUTOEXEC.BAT** írásmód a tulajdonnévi bizonytalanság elfedése a kiemelés eszközével. Nem indokolt (és nem is esztétikus) a szavakat végig nagybetűsre írni, az csak a valódi betűátírási hibák jár, amikor nem „kijönnek” a szót, hanem „lebetűzünk”.

3. Az **Autoexec.bat** forma gyengéje az, hogy egyes fájlnevek inkább köznévi szöként használatosak, de talán ez az a pont, ahol a sokféle szempontnak mégis kompromisszumot lehet kötni.

4. Az **Autoexec.Bat** írásmód feltehetően abból fakad, hogy a pont után nagybetűt kívánnak, annál is inkább, mert az csak a véletlen műve, hogy az elhatárolás jelölését a programozók annak idején a pontra definiálták, de az bármi más lehetne, általános nyelvi logika szerint például szóköz. A fájlnevet két fele egyébként is úgy viszonyul egymáshoz, mint a keresztnév és a vezetéknev, ahol a kiterjesztés megfelel a családnévnek, az első (lehetőséges) 8 karakter pedig a személynévnek.

5. Az **autoexec.BAT** írásmódot helyesírási normává tenni nem sok értelme lenne, mellette szóló érvek csak a kiterjesztés vizuális kiemelésével elérhető gyorsabb azonosítás hozható fel.

6. Az **AUTOEXEC.bat** írásmód szándéka talán azonos az előző pontban említett változattal, de hatása sokkal gyengébb, mert az első tag rendszerint hosszabb, s nagybetűs írás esetén minél

több betűből áll, annál inkább csökken az azonosítás sebessége.

Kiterjesztett szárnyak

A fájlnevek a kiterjesztést tartalmazó fele önálló életre (szárnyra) kelt, jelzős szerkezetek gyakori szereplőjévé vált. A Vírushatározóban gyakran előfordul például az, hogy a vírus milyen állományokat fertőz meg. Azt az írásmódot választottuk, hogy „COM fertőző vírus, EXE fertőző vírus”. De ezzel kapcsolatban bőven maradtak bennünk kétségek.

— Az első a kiterjesztés előtti **pontra** vonatkozik. Hogyan lenne helyesebb írni: az orientáló funkciót betöltő ponttal vagy annak elhagyásával?

— A másik, hogy végig nagybetűs legyen-e a kiterjesztés vagy végig kisbetűs? Esetleg egy kezdőbetűs?

— A harmadik gond a kiterjesztésnek és a jelzett szónak az összekapcsolása: maradjon-e szóközzel elválasztva, vagy legyen helyette kötőjel. Ha kisbetűsre végződik a kiterjesztés, akkor pedig írhatjuk-e esetleg egy szóba is?

Itt most nem megyünk bele a részletes elemzésbe. Mi az említett könyvben leírtak ugyan a vokst, de az Alaplapban sem mindig ugyanúgy „szavazunk”, és nem is vagyunk meggyőződve arról, hogy miénk a jó megoldás. Várjuk ez ügyben is mindazok véleményét, akikben kialakult egy akár szakmai, akár nyelvészeti érvekkel alátámasztott állásfoglalás. Mankóként ehhez egy adott példával felsoroljuk, hogy mit is kell mérlegelni: EXE fertőző, .EXE-fertőző, .exe fertőző, exe-fertőző, .exe-fertőző, .EXE-fertőző, .EXE-fertőző, EXE-fertőző, exe-fertőző, exe-fertőző, exe-fertőző, EXE-fertőző, EXE-fertőző, EXE-fertőző, EXE fertőző, exe-fertőző, exe-fertőző, exe fertőző, EXE-fertőző, EXE-fertőző, EXE fertőző.

A bemutatott 16 változat közül kellene kiválasztani és használatra javasolni a legelfogadhatóbbat. A jelenlegi helyzetben ugyanis a szakmai anyagokban szinte mindegyik előfordul, ami bizonyára csak jelentéktelen megértési problémát okoz, de akkor sem válik a magyar számítástechnikai szakma nagy dicsőségére. Ráadásul éppen egy olyan szakmáról van szó, amely a program-szövegek szintaxisában viszont teljes precizitást követel meg, s egyetlen karakter elmulasztása is (lásd szöveget Marszonda) hatalmas károkat tud okozni. A programozás által követelt pontosságából valami töredéknyi igazán kisu-gározhatna a szaknyelvre is!

Faklen Pál

Nyelvi morfondírozások

Multi*média* vagy multi*médium*?

Az elmarasztaló glosszák időnkénti felbukkanása és a sajtó egy részének „médiumplántáló” igyekezete ellenére, a tömegtájékoztatási eszközökben, szakmai kiadványokban és a köznap beszédében is tovább tartja hadállásait a média kifejezés — nem tisztelvén kellőképpen a latin többes számot. Miután a médium kontra média per a multimédián és több más szakkifejezésen keresztül közvetlenül érinti a számítástechnika nyelvét is, szándékaink szerint sorozatként indított nyelvi morfondírozásaink első darabjaként ezt a témát választottuk.

Aki netán még nem találkozott volna a *média* szót támadó vagy védő nyelvész-kedő vitacikkkel, annak is észre kellett vennie a sajtó körüli csatározások kétféle szóhasználatát: az egyik tábor médiumokat, a másik médiákat emleget, de nem politikai elkötelezettség alapján, hanem tisztán nyelvi megfontolásokból. Vajon melyik félnek van igaza? Az egyik szemléletmód latin nyelvi szabotázstól kér számon a magyar nyelvtől, a másik elfogadja a formális logika szabályait időnként nem tisztelő nyelvfeljókodás tényeit.

„Szellemidézésre foglalt!”

Mindenekelőtt talán azon kellene elgondolkodni, hogy a spontán szóhasználat a kommunikációs eszközökre miért a *média* alakot részesítette előnyben, hiszen a fogalom megjelenésekor már ott állt készenlétben az egyes számú latin alakból meghonosodott *médium* is. Hát igen, éppen ez az első bökkenő. A *médium* szóhárszázadok előbb rátelepedett egymáshoz elég közel álló három másfajta jelentés:

— Hipnotizált és hipnotizálható személy.

— Valamely ügy eszközként felhasznált szereplője.

— Spiritiszták túlvilági közvetítő alanya. („A” médium.)

Emellett a *médium* szónak a latinban meglévő általános tárgyi, illetve fogalmi értelmű *közvetítő eszköz* jelentése nálunk nem akart gyökeret ereszteni.

És akkor jött a *média*... Annak megsikerült! Hogy is van ez?

A magyar nyelv új fogalmak nyelvi köntösét keresve nem nagyon szereti a már lefoglalt jelentésű szavakat. Lehet vitatkozni azon, hogy ez jobb vagy rosszabb annál az angol szokásnál, hogy ott a szó előszeretettel aggat magára megannyi jelentéstartalmat, de tény, hogy a mi nyelvünk kifejezetten szenved, ha időnként ilyesmivel traktáljuk. Talán a *médium* szó korábbi tartalmának mindig *személyre* utaló jellege is taszította az egészen más szférákban mozgó fogalomrendszer.

Annai bizonyos, hogy bár a *média* szó sokáig csak szűkebb szakmai körben volt használatos, mégis általános jelentésszerű különbség alakult ki a kettő között. Egészen mást jelent a *médiumkutatás*, mint a *mediakutatás*, vagy a *kísérleti médium*, mint a *kísérleti média*. *Médiumtörvényt* is lehetne hozni (a spiritiszta médiumok működéséről), és az valószínűleg könnyebben is készülné el, mint a *mediatörvény*. Az okkultizmus felszínre törése köznap beszédünkben is teret adott a *megfelelő médium kiválasztásának*, amit az érintettek (bármelyik oldalon) aligha szeretnének összekeverni a *megfelelő média kiválasztásának* műveletével. Szemantikai oldalról nézve tehát a *médium* és a *média* eltérő jelentések hordozójává vált, s ez önmagában is megerősítette a szóalakok elkülönülését.

A tájékoztatói eszközöknek *média*-ként való megnevezését általában nem

is jelentéstani, hanem (latin) nyelvtani megfontolásokból támadják, amiért a latinul már többes számú értelmet nyert *média* alakot helytelen módon egyes számban értelmezzük, és a *médiák* formával duplán tesszük többes számba. Csak hát a *média* már nem *media*! A más nyelvekből átvett szavak a befogadó nyelvben mindig önálló életre kelnek. Elvesztve eredeti formájukat, alkalmazkodnak a befogadó nyelv hangrendszeréhez (mert másként nem lehetnek), s elveszítik a ragozásukra, képzésükre vonatkozó korábbi nyelvtani szabályokat, hiszen minden nyelvnek önmagával konzisztens rendszert kell alkotnia, s a jövővényeknek nem a küldő nyelv szabályai szerint kell viselkedniük, hanem a befogadó nyelvét szerint.

Dupla vagy semmi

A *média* szó a magyar nyelvi környezetbe átlépve és ott meghonosodva már egyáltalán nem képes érzékeltetni eredetileg többes számú mivoltát, ezért sem okoz zavart, amikor *médiákat* emlegetünk. Ha magyar beszédünkben megpróbálnánk kiiktatni mindazokat a nyelvi formákat, amelyek más nyelvekből származnak és amelyeknek általunk történő használata nyelvtanilag nem követi az átadó nyelv szabályait, akkor valószínűleg lépten-nyomon megakadna a beszédünk. Jövővényszavaink túlnyomó többségére éppen az jellemző, hogy az eredeti szóalakhoz képest módosulva (sőt csonkán vagy megtoldva) honosodnak meg, mert csak ezen az áron tudnak beilleszkedni hangrendszerünkbe, nyelvtanunkba. Lehet, hogy ez illetlenség az átadóval szemben, de ettől még tény, sőt törvényszerű folyamat, amely a többi nyelvben is hasonlóképpen játszódik le.

A *média* szó „viselkedése” nem valami rendkívüli jelenség, de legközelebbi rokonai már annyira szervesen beilleszkedtek, hogy észnünk sem jutna származásukat — és többes számú alakjukat — firtatni. Íme néhány a változatlanul vagy magyaros írásmódban átvett szavak közül, amelyek eredeti latin többes számú alakját egyes számban használjuk, és gyönyörűen tehetjük saját nyelvi logikánk szerint

„még egyszer” többes számba: *nulla, opera, orgona, széria, váza, tufta, textília, fekália, parazita*. Ebből a felsorolásból nem lóg ki a média.

A média kifejezés egyes számú főnévként való elfogadását valószínűleg elősegítette egy formális analógia is (ami persze egészen más nyelvi és nyelvtani jelenség), az ugyancsak egyes számú alakban „-ia” végződésű, többnyire görög eredetű és elvont vagy gyűjtőfogalmat jelölő szavak nagy száma és gyakorisága. (Anomália, analógia, akadémia, biblia, frekvencia, fúria, mánia, memória, melódia, komédia, energia, kalória, fantázia, filozófia... és a sokféle -ológia.)

Rébuszokban beszélve

A vitákban a médium-pártiak időnként európaiságunkra szoktak apellálni. Pedig a *média* szó átvételében talán a nemzetközi gazdasági vérkeringésbe való bekapcsolódásunk is nagy szerepet játszott. A szakmai kifejezéseket logikább orientáló angol nyelv ugyanis a saját testében is idegen *media* többes számú alakot széles körben terjeszti, az egyes számú *medium* alakot pedig csak speciális esetekben, egészen ritkán használja. Szösszestételekben, jelzős szerkezetekben mindig a többes számú forma szerepel (*media research* = médiakutatás, *media planning* = médiatervezés, *media owner* = médiatulajdonos stb.), akkor is, ha annak magyar megfelelője egyes számú szóalakot követel meg. Mindez a *média* kifejezés egyes

számú értelemben való meghonosodásának malmára hajította a vizet.

Egyre többször találkozhatunk azonban a *média* többes számú értelemben való használatával is. Pedig a magyar nyelv a *média* szót aligha tudná többes számúként kezelni, hiszen akkor olyan mondatokkal kellene megbirkózni, hogy például „A vidéki média *foglal-kozzának* inkább a helyi ügyekkel”. Az ige gyorsan leleplezi az össze nem illőséget, birtokos szerkezetet használva azonban áthatolhatatlan fogalmazási homály keletkezik. Ha azt mondjuk, hogy „a média irányításában személyi ellentétek vannak”, az vonatkozhat a tájékoztatási eszközök összességére éppúgy, mint egyetlen újságra vagy a televízióra, s az sem biztos, hogy a szövegtörzset alapján egyértelműen el lehet jutni a helyes megfejtésig. Pedig, ugye, Kosztolányitól már megtanultuk, hogy az olvasót bizonytalanságban hagyni egy pillanatra sem szabad!

A címben szereplő *multimédia* is okozhat egyeseknek némi fejtörést. Egyrészt nagyon valószínű, hogy e szó nálunk ebben a formában terjed el, hiszen tömör, jól hangzik, beilleszthető a magyar nyelvbe, és az ékezetet leszámítva az egész világon így használják. Helyette mást javasolni (és elfogadtatni) elég nehéz lenne. A *multimédium* bevezetése a médium-pártiakat saját okfejtésükkel állítaná szembe, mert a multi az összetétel másik felére vonatkozóan sokszorosát jelent, akkor pedig az hogy állhatna egyes számban? A

multimédiumok sem megoldás, mert a fogalomhoz egyes számú főnevet is kellene alkotni, a csak többes számú (a magyarban egyébként is igen ritka) alak kezelhetetlen lenne. Ráadásul a *multimédia* olyan gyűjtőfogalom, amely csak egészen kivételesen kaphat többesjelet, tehát ritkán lenne alkalom a „többes-többes” miatti háborgásra.

Mit mond az írás?

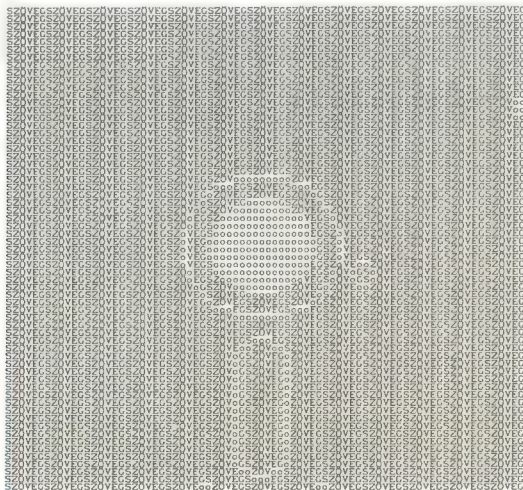
Valószínűleg mindenkinek bele kell törődnie a *média* szó (vagy csak a multimédia?) meghonosodásába, annak üsz-szes nyelvhasználati következményével együtt. Erre a pontra a nyelvészek már korábban eljutottak, s érdemes az MTA Nyelvudományi Intézet által 1985-ben megjelentetett kétkötetes *Nyelvművelő* kézikönyvben erről a témáról közölt szócikket teljes egészében idézni:

„Mivel a latin eredetű *médium* ma elsősorban szuggerálható, hipnotizálható, ill. könnyen befolyásolható személyt jelent, s csak másodsorban közvetítő eszköz, tényezőt, közeget, bizonyos társadalomtudományok, a hirdáts-technika és a reklám szaknyelvében angol mintára elterjedt a szó latin többes számú alakja, a *média*. Összefoglaló megnevezése a tömegkommunikációs, ill. reklámszövegeknek (sajtó, rádió, tv, plakát, hirdetés, reklám stb.).

Gyakran többes számban is használják: *médiák*. Ez voltaképpen szabálytalanság, mivel a már amúgy is többes számú szóalak kapott újabb többesjelet (mint pl. az *Alpések*, *Beatlek*): a legjelentősebb *médiák*; *reklámmédiák*; stb. A szó idegen volta és a *médium*-tól való jelentésbeli elkülönülése folytán azonban ma már általában nem ismeretes, hogy latin többes alakról van szó, s így ez a „szabálytalanság” aligha lehet zavaró.” (II. kötet, 111. oldal.)

Tegyük hozzá kiegészítésként, hogy azóta a *média* szó jelentéstartalmát meghatározó sajtó, reklám és hírközlés mellé a számítástechnika is gyors ütemben felzárkózik, bár a köznyelvbe ez még nem nagyon szivárgott át. (Talán a multimédia lesz az első „betolako-dó”.)

A vita persze a nyelvészeti szakvéleményektől függetlenül a gyakorlati nyelvhasználat „népszavazásán” dől majd el, s ha a *média* jobban megfelel a magyar nyelv belső törvényszerűségeinek, akkor bármilyen nemes szándék is vezeti a másik felfogás képviselőit, hiába „kötik a médiumot a karóhoz”.



Akár életfogytiglanit is kaphat!

100 szóznak mennyi (lesz) a vége?

Igaz ugyan, hogy nyelvi kérdésekben nem népszavazásokon szoktak dönteni, de tulajdonképpen mindig a nép dönt. Még csak nem is (választott) képviselőin és (önjelölt) szószólóin keresztül, hanem a lehető legközvetlenebb módon: ahogy anyanyelvét használja. De azért nem árt néha megkérdezni a népet, hogy miként vélekedik bizonyos dolgokról. Az Alaplap, mint ezen összeállításával a számítástechnikai szaknyelvben érdekelt nép (önjelölt) szószólója, most ilyesmit kezdeményez. Játéknak tűnhet, amire olvasóinkat kérjük, de csak azoknak ajánljuk, hogy részt vegyenek benne, akik minden játékot nagyon komolyan tudnak venni.

„Pofonegyszerű” feladatot találtunk ki. (Olyan értelemben, hogy egyszerű volt kitalálni, és az lenne a pofon, ha nem érkezne rá elég válasz.) A résztvevőktől „mindössze” azt kérjük, hogy küldjenek be 100 szót az Alaplap szerkesztőségébe (1441 Budapest VIII., Reguly Antal u. 8.), lehetőleg mágneselemre tett egyszerű szövegállományként. (A lemezt visszaküldjük.) A 100 tulajdonképpen 5 x 20, mert minden felsorolt kategóriában legalább huszat kell összegyűjteni. A listában szerepelhetnek önálló szavak, eltérő szóalakok és összetett kifejezések is, de valamennyit a számítástechnikai szaknyelvből kell meríteni.

A csoportosítás ismérvei:

1. A legjobban sikerült magyarázatok.
2. Magyar írásmódban megtartandó idegen szakkifejezések.
3. Eredeti írásmódban megtartandó idegen szakkifejezések.
4. Száműzendő és magyarral helyettesítendő idegen szakkifejezések.
5. A szaknyelvből kigyomlálandó „vadmagyarázatok”.

Természetesen a 20—20 tétel csak a megoldás elfogadásának minimális követelménye, amelyetől felfelé bátran el lehet térni. Nem beküldési feltétel ugyan, de a 4. és 5. kategóriában kívánatos, hogy a beküldő írja le a helyet-

tesítésre általa leginkább alkalmasnak tartott szavakat és kifejezéseket is.

A beérkező anyag feldolgozásából első sorban azt szeretnénk megtudni, hogy olvasóink mennyire ítélnek meg velünk azonosan vagy tőlünk eltérő módon a szaknyelvi, szóhasználati kérdéseket. Másik célunk, hogy friss ötleteket merítsünk a számítástechnikai szaknyelv csiszolásához és az új (vagy akár régi) szakmai fogalmakat legjobban kifejező nyelvi megoldásokhoz. Lapunkban természetesen közreadjuk az összesítés eredményét is, mágneselemmel mellékletünkre pedig — tanulságként és ötletadónak — szeretnénk a javaslatok özönét rátenni, amhez csupán annyi kell, hogy bennünket is elárasssanak a megoldások.

Hogy vállalkozókedvűket valamivel serkentésük is, a 25 legjobb anyag beküldje közül háromnak kisorsolunk egy nem akármilyen jutalmat: Örökös Tiszteletbeli Olvasók lesznek, vagyis az Alaplapot életfogytig ingyen kapják. (A nyertesek élete, illetve az Alaplap élete végéig!)

A beküldési határidő:

1992. október 30.

Szó és szöveg

Angol nyelven

Style counsel. (Riva Grammatik IV, Editor Software StyleWriter és RightSoft RightWriter helyesírás-ellenőrző szoftvercsomag.) PC User, 1990/6.

Digital Dharma. (Az ősi tibeti irodalom számítógépes rögzítése.) Electric Word, 1990/6. Could WordPerfect Rhymer be finer? (Diákok nyelvtanulását segítő, 93.000 szavas amerikai-angol rímestár a WordPerfect szövegszerkesztőhöz.) Byte, 1990/11.

The multilingual edge. (A számítógépes fordítás távlatai.) Byte, 1991/3.

Evaluating Computer Generated, Domain-Oriented Vocabularies. (Számítógépes szakszótárak gépi generálásának fejlődése — természetes nyelvű információ visszakeresése.) Information Processing & Management, 1990/6.

Text checkers: mind your grammar. (Hat szövegellenőrző, helyesírás-ellenőrző program bemutatása és értékelése.) What Micro?, 1991/7.

Can a grammar and style checker improve your writing? (Kilenc — Macintosh és MS-DOS rendszerekre írt — helyesírás- és stílusellenőrző programtermék bemutatása.) Byte, 1991/8.

Write on target: 15 writer's tools. (15 különböző típusú szövegellenőrző programcsomag bemutatása és értékelése.) PC Magazine, 1991/9.

Software reviews: Language translation program. (Az angolról spanyol nyelvre történő fordítást segítő, IBM kompatibilis gépen futó, könyvtári alkalmazásra ajánlott Spanish Assistant szoftver.) Library Software Review, 1991/május-június.

DM-Dialog. (Kisérleti nyelvi fordító rendszer.) Computer — IEEE, 1991/6.

Parallel access to an English dictionary. (Öt transzputerre elosztott angol nyelvű számítógépes szótár párhuzamos elérése.) Microprocessors and Microsystems, 1991/július-augusztus.)

Német nyelven

Übersetzungsprogramme: man spricht Deutsch. (Idegennyelvi fordítóprogramok összehasonlítása és értékelése.) Computer Persönlich, 1991/2.

Einfach und gut: PC-Wörterbuch von Langenscheidt. (Langenscheidt német-angol/angol-német PC-szótár.) BIT — Büro und Informationstechnik, 1991/4.

Fehlerlos schreiben? (Helyesírás-ellenőrző és -javító programok tesztelése.) PC Praxis, 1991/5.

Rechtschreibprogramme unter der Lupe. (Helyesírás-ellenőrzővel rendelkező szövegszerkesztő programok összehasonlítása.) Chip, 1991/8.

Ways — der neue Windows-Knüller? (Ways for Windows online fordító és helyesírás-ellenőrző program tesztje.) PC Praxis, 1991/8. Vorstellung: Fünf Übersetzungsprogramme. (Természetes nyelvű fordítóprogramok bemutatása.) PC Welt, 1991/11.

MADE-INFO Kft.



**15 000
PÉLDÁNY**



Az egyre sokrétűbb információtechnikat az őszi katalógusunkban még magasabb szinten foglaljuk össze. Ezzel segítjük mindazokat, akik e területek szolgáltatásaiból választani, illetve termékeiből vásárolni kívánnak.

Katalógusunkat saját, aktualizált címjegyzékünk alapján 10.000 felhasználóhoz továbbra is **INGYENESEN** juttatjuk el.

Katalógusunk ismét tartalmazza a kedvelt **TEMATIKUS TÁRGYMUTATÓ**-t. Újdonság a különálló, borítós **TELEFONKÖNYV**.

EUROPA TELECOM '92
Az október 12 - 17 között megrendezendő távközlési világtalálkozó alkalmából a távközlést kiemelten kezeljük, ezért azt külön katalógusban jelentetjük meg.



Partnereink katalógusunk és adatbankunk alapján vásárolnak: Ha szeretné, hogy naprakész információt adhassunk Önökről is, árváltozás, termékbővülés, címváltozás esetén küldjön tájékoztatót, illetve adjon prospektust az üzletkötőnknek.

KEDVEZMÉNYEK : AZ ÖSSZES ÁRBÓL

- ♥ **3%** jár azon cégeknek, akik az INFORMÁCIÓTECHNIKA '91-ben és az INFO-KATALÓGUS '92 I. félévi számában már szerepeltek, továbbá a II. félévibe is jelentkeznek.
- ♥ **7%** mindazoknak, akik a teljes szerződött összeget a szerződés kötésekor befizetik.
- ♥ **10%** illeti meg azon cégeket, akik színre bontott, tűkőrméretű anyagot, és a hozzátartozó chromalin próbát vagy színes nyomatot adnak le.

LAPZÁRTA:
augusztus 11.



MEGJELENÉS:
október eleje

Katalógusunk nemcsak budapestieknek készül; több ezer **VIDÉKI** cég és önkormányzat ingyenesen megkapja!!!



**MEGRENDELÉSÜKET AZ ALÁBBI CÍMEN, ILLETVE TELEFAXSZÁMOKON VÁRJUK
MADE-INFO Kft.**

1476 BUDAPEST, PF.110.
178-4421, 227-3647

- ▷ Vállalkozói Kamara tagjai
- ▷ Bankok
- ▷ Biztosító társaságok
- ▷ Önkormányzatok
- ▷ Országos hatáskörű szervek
- ▷ Szerkesztőségek
- ▷ Egyetemek, főiskolák
- ▷ Nyomdák
- ▷ Követsegek kereskedelmi képviselői
- ▷ Gumí-
- ▷ Kőolaj-
- ▷ Gyógyszer-
- ▷ Műanyag-
- ▷ Festék-
- ▷ Kozmetika-
- ▷ Gabona-
- ▷ Tej-
- ▷ Cukor-
- ▷ Dohány-
- ▷ Szesz-
- ▷ Bor-
- ▷ Sör-
- ▷ Hús-
- ▷ Konzervipari cégek
- ▷ Kárnyszervédelmi cégek
- ▷ Számítás-
- ▷ Biztonság-
- ▷ Irodatechnikai cégek
- ▷ Irodabútor forgalmazó
- ▷ Távközlési cégek
- ▷ Autómárka kereskedések
- ▷ Nyelviskolák
- ▷ További oktatási intézmények



Ha Önnek bármilyen eredetű ill. verziójú ÉkSzer szövegszerkesztője van, hozzá be hozzánk augusztus 31-ig, és az aktuális ár 30%-áért Ön egy jogtiszta 5.05 verziójú ÉkSzer szoftver tulajdonosa lesz. Így jogosulttá válik az ezzel járó garanciális és szoftverkövetési szolgáltatások igénybevételére.

ÉkSoft Kft. Budapest 1068 Szófia u. 8.
Tel/fax: 122-3973

DATENTECHNIK

Kereskedelmi Képviselőt
1016 Budapest I., Naphegy tér 8.
Tel./Fax: 175-0182 • Telex: 22-4800

Digitális átviteltechnika, speciális távközlési rendszerek, integrált hangú és adatsatormás készülékek (DOV), professzionális modemek nagy választékban (WAN) postai engedéllyel, csomagkapcsolás, Ethernet-Token Ring helyi hálózatok (LAN), különböző hálózatok integrálása, minőségi PC-k. Komplet hálózatok tervezése, telepítése, egyedi megoldások.

SZÁMÍTÁSTECHNIKA KULCSRAKÉSZEN!

SZÁMÍTÓGÉPEK, NYOMTATÓK, MODEMEK SZÉLES VÁLASZTÉKA:

- AT, 386, 386SX, 486 számítógépek minden kiépítésben. (3 év garanciával)
- Laptop, notebook gépek.
- EPSON, STAR, NEC, HP nyomtatók teljes választéka.
- DISCOVERY és US ROBOTICS modemek és táv-adatviteli rendszerek.
- APC szünetmentes tápegységek.
- SZOFTVEREK és SHAREWARE-ek teljes választéka.
- NOVELL HÁLÓZATI SZOFTVEREK, hálózatiptés.
- Számítógépek és tartozékok javítása.

AT számítógép: 20 MHz, 1 MB RAM, 1,2 MB floppy,
40 MB winchester, monokrómonmonitor: 53 900,- Ft+ÁFA
Készpénzért: 51 200,- Ft+ÁFA

3 ÉV GARANCIÁVAL!

Kérjük, telefonáljon vagy írjon, és mi örömmel adunk felvilágosítást, küldünk részletes árjegyzékelt!

QWERTY

High Tech Kft.

1117 Budapest XI., Orly u. 4.
Telefon: 166-3098, 185-2687, Fax: 185-2687
BBS: 11-87-950 BUDAPEST BBS

NE FELEDJE: Nevünk ott található
az Ön számítógépének billentyűzetén is!

Milyen típusú hálózatra van szüksége?



ARCNET, ETHERNET, IBM TOKEN RING?

Keressen fel bennünket, és mi segítünk a választásban.
Számítógép-hálózatok:

- tervezése,
- kiépítése,
- installálása,
- bővítése.

24 havi garanciával, kedvező áron.

Gyorsaság, megbízhatóság
és kompatibilitás mindenekelőtt!

AST számítógépek
BEST modemek
EPSON nyomtatók és kiegészítők
HEWLETT PACKARD perifériák és tartozékok
QUANTUM és WESTERN DIGITAL winchesterek

Szeretettel várjuk Önt
szaküzletünkben



UNITRADE
Szervezési, kereskedelmi
és Számítástechnikai
K.F.T.

1073 Budapest VII., Erzsébet krt. 48.
Telefon/Fax: 142-2115

...nemcsak számítástechnika

HETENTE FÖLDKÖZELBEN

a **TELEHOLD**

ÖN A LEGTÖBBET KAPJA,
ha megrendeli a hazánkban fogható valamennyi
fontos műholdprogram legrészletesebb műsorfűzetét.
Ingyenes hirdetési lehetőség,
a hazai és a környező országok tévéműsorai.

Keresse szerdától az újságárusoknál
vagy fizessen elő!

Egy évre előfizető olvasóink 12 hétén át ingyen kapják

a **TELEHOLD** at!

OFFI-COMP Kft.
1062 Budapest, Bajza u. 52.

Azonnali szolgáltatásaink:

- Műszaki, kereskedelmi és jogi szövegek fordítása,
- Szövegszerkesztés,
- Kiadványszerkesztés

megbízhatóan kiváló minőségben

minden európai nyelven!

Megrendelés telefaxon is!

telefon/fax: 1320-729

Rovatunkban hónapról hónapra, egymásra épülő ismeretekkel „bombázva” az olvasókat, betekintést adunk a mesterséges intelligencia kutatásának, az erre alapuló fejlesztésnek, alkalmazásnak a fejezeteibe.

Leginkább átfogóan kapcsolódik össze az elmélet és a gyakorlat a szakmai köztudatban szakértő rendszerekként „tisztelt” MI-fogalom — és természetesen a konkrét produktumok — terén.

Innen ugyan ismét elágazik majd a diszciplínákat is számba venni hivatott sorozat tematikája, de az e havi téma segít közel hozni az eddig tárgyalt anyagot azokhoz a szakemberekhez, akik a dolgokat szeretik saját „hagyományaikhoz” is visszavezetni — vagy fogalmazhatjuk fordítva: szívesebben nyújtózkodnak az újabb irányok felé, ha lábukat a saját „házuk táján” vethetik meg...

Most következő cikkünk tehát részint felépítménye az eddigi írásoknak, melyek egyeseknek talán túlságosan elvontnak tűntek, részint alapul szolgál a kapott ismeretek célirányos rendezéséhez, vagyis a továbbiak — kinek-kinek a maga szakterületén hasznosulható — befogadásához.

A tudás minden előtt!

A számítógépes programozás történetének egyes szakaszait az ott alkalmazott programstruktúrák stílusokkal, programozási paradigmákkal jellemezhetők. Az egyes szakaszok ugyanis rájuk valló módon válaszolnak azokra a szoftvertechnológiai kérdésekre, hogy például: milyen program „jól struktúrált”; milyen adat- és algoritmusábrázolási módokat kövessünk; milyen technológia (elmélet, módszertan, eszközkészlet) áll az alkalmazásfejlesztők rendelkezésére, stb. Az alábbiakban ilyen megközelítéssel vázoljuk a szoftvertechnológia — számos egyéb szempontból tekintve is vizsgálható — fejlődésének három legfontosabb szakaszát, bővebben tárgyalva a tudáslapú technológia alapkérdéseit. Az ismert — jól ismert — fogalmak és összefüggések mondhatni „visszaköszönnek”, de most a viszonylatok megváltozására kell igazán figyelni.

Algoritmikus programok

Milyen szerkezetű egy hagyományos, algoritmikus program? Mindenki számára világos az 1. ábráról leolvasható képlet, amely szerint egy ilyen program kódja a programvégrehajtáskor passzív adatrészből, és az ezen dolgozó — vagyis aktív — algoritmusrészből áll. Az adatok között kiemelt szerepük van az input és az output adatoknak.

Egy hagyományos program célja tulajdonképpen egy függvény megvalósítása: „leképezni” az input adatokat a nekik megfelelő output adatokra (mint egyetlen adekvát megoldás együttese). A leképezés módját, HOGYAN-ját a program algoritmusrésze fogalmazza meg. A programhoz szervesül a program belépési pontja(i)t tartalmazó rész is, amely(ek) révén indíthatja el a felhasználó a programot.

Az ilyen programok írására egyre magasabb szintű (algoritmikus vagy procedurális) programozási nyelveket hoztak létre, különböző programfejlesztési módszereket, rendszerfejlesztő környezeteket dolgoztak ki. Idővel azonban egyre több olyan programot kellett írni, amelyek nagyon sok, a háttértárakra kihelyezett adattal dolgoztak. Az 50-es években megjelent a fájl (adatállomány) fogalma. Számos fájlkezelő rendszer született, biztosítva a fájlokban tárolt adatok hatékony kezelését (beírás, módosítás, törlés, rendezés stb.). A fájlok, illetőleg a fájlkezelő rendszerek lényegében a programok adat- és algoritmusrészeinek kiterjesztéseként tekinthetők.

Adatállomány, -bázis, adatbázisrendszer, -alkalmazás

Az elmúlt 30 év során az egyszerű fájlkezelő rendszerek általánosítása-

ként jöttek létre az adatbázis-kezelő rendszerek. Ezek egyszerre több felhasználó és többféle alkalmazói program számára központi helyen és redundanciamentesen tárolva biztosítják az adatok kezelését. Az adatbázis-kezelő rendszerek a nagy, (el)osztott adatbázisok kezeléséhez hatékony mechanizmusokat, kényelmes szolgáltatásokat nyújtanak. Program helyett itt adatbázis-alkalmazás(ok)ról beszélünk (ad hoc igényeket a végfelhasználó közvetlenül is megfogalmazhat). Mint a 2. ábra is mutatja, az adatbázis(ok) a program adatrészeinek, míg az adatbázis-kezelő rendszer az algoritmusrészeinek kiterjesztéseként funkcionál(nak).

Az adatbázis-technológia alkalmazása „kivette” az általánosan használt adatokat és kezelésüket az alkalmazói programok írói és a végfelhasználó kezéből. Az 1. ábrán látható képlet a következőképpen módosul: az adatállománnyal kibővített adatrészt továbbra is passzív. Az algoritmikus rész helyét az adatbázis-kezelő rendszer, valamint a megengedett adatbázis-kezelő nyelven megírt alkalmazói programok foglalják el, azonban nem csak ezek. Egy adatbázis-alkalmazás ugyanis „kinyílt” a végfelhasználó felé: olyan — egymással kombinálható — szolgáltatásokat kínál, amelyekkel az feladatokat fogalmazhat meg, és adhat oda a rendszernek megoldásra. Még egy markáns külön-

ség: e szolgáltatásokkal mind a fejlesztő, mind a végfelhasználó lényegében nem valamely algoritmus megadásával, hanem leíró stílusban deklarálhatja a rendszer számára azt, hogy MIT, azaz milyen feltételeket kielégítő adatokat kíván outputként megkapni. Az algoritmikus programozás mellett megjelent tehát a deklaratív feladatleírási stílus is, amely jó néhány programozási nyelvre (például Lisp, Prolog) is jellemző, és amely természetes módon valósul meg, az ún. tudásalapú rendszereknél.

Az adatbázis-alkalmazások célja legelőször tér el a hagyományos programtól: az adott adatbázis(ok) nagyon sok alkalmazói programot szolgáltatnak ki, továbbá a rendszer nyitottsága a végfelhasználó felé sokféle (ad hoc) funkciót biztosít. Indítása nem explicit módon kijelölt belépési pont(ok)tól kezdhető, hanem egy lekérdező nyelvet használva tudja a felhasználó előírni, hogy MIT akar. Ami még lényegesen módosult: (például relációs adatbázis-alkalmazásoknál) egyetlen lekérdezés során mind az input, mind az output adatok lehetnek halmazok is, sőt, a lekérdező nyelven „programozó” halmazműveletekkel is manipulálhat.

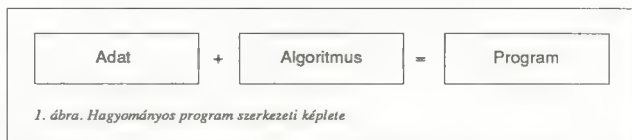
A programszervező, programtervező, programozó és végfelhasználó mellett kiemelkedik egy igen fontos új szereplő, az adatbázis-adminisztrátor. A működő rendszernek először lesz ilyen értelemben összetevője az ember: adatbázisrendszer = adatbázis + -kezelő szoftver + -adminisztrátor

Időközben új szabványok és nyelvek jelentek meg: adatmodellező szabványok, adatbázisnyelv(-)szabványok (pl. Sequential Query Language, SQL), negyedik generációs nyelvek (4GL: pl. SQL*Forms), majd 1985 után a CASE (Computer Aided Software Engineering) technológia első változatai. Rohamléptekben készültek a különböző adatbázis-alkalmazások. 1988-ra már 400-nál több adatbázis-kezelő rendszer volt a piacon. A három klasszikus adatbázismodell (a hierarchikus, a hálózatos és a relációs) közül a 70-es évek közepétől a relációs modell terjedt el leginkább.

Az integráció hamarosan elindult: az adatbázis-kezelő rendszerek és programozási nyelvek egymásból hívhatósága ma már természetes igény, a piaci pozíciót erősen befolyásoló tényező.

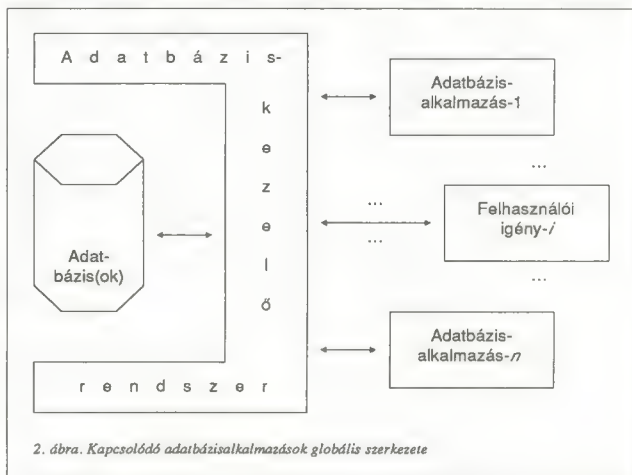
Tudásalapú rendszerek

A MI-kutatások célja lényegében az intelligens emberi viselkedés megértése, s ennek alapján „intelligens” számítá-



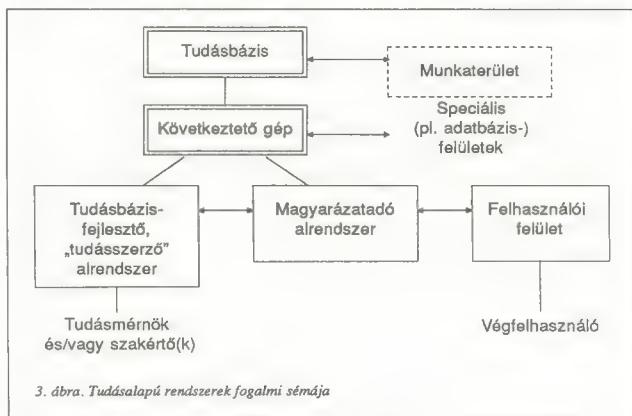
tógépes programok létrehozása. Fel kellett ismerni: a problémamegoldás képessége lényegében attól függ, hogy mennyi, és milyen magasan kvalifikált

minőségű specifikus ismeretanyagot képes mozgósítani a problémamegoldás során. A 80-as évek elején tehát különböző tudásábrázolási módszere-



ismeretanyag — tudásbázis — áll rendelkezésre az adott tárgyerületről („a tudás elve”). Következésképpen egy program erőssége, intelligenciája is attól függ, hogy milyen sok, és milyen

ket és (az így ábrázolt ismeretanyagot „működésbe hozó”) problémamegoldó, következtető mechanizmusokat fejlesztettek ki — specializált tudásalapú rendszerek építése és működtetése céljából.



A tudásalapú rendszereknek a szerkezete sem a megszokott. Az alapvető itt: a tárgyerületi ismeretek különüljenek el a rendszer többi részétől, így a (hatékony megoldáskereső algoritmusokat tartalmazó) problémamegoldó ismeretektől is. Előbbi tudásbázisnak, utóbbi pedig következtető gépnek nevezzük (l. 3. ábra). Az ilyen szerkezetű rendszereket tudásalapú rendszereknek nevezzük. (Az első, bonyolult valós problémák megoldását célzó tudásalapú rendszereket szakértői rendszereknek hívták, azonban ez a megnevezés sok félreértés forrása volt, így ma egyre kevésbé „divatos”).

Az emberi problémamegoldási folyamatot elemezve mindenki megállapíthatja maga is, hogy benne nem (előre beprogramozott) algoritmusok működnek, hanem egy probléma megoldásakor apróbb, önmagukban is értelmes tudáselemekkel, ismeretdarabokkal manipulálunk: ökölszabályokkal (például: „Ha felkapcsoláskor nem ég a villany, akkor meg kell nézni, nem égett-e ki a villanykörte.”), „tudáskeretekkel”, frame-ekkel/sémákkal (gondoljunk például a magasabb rendű fajok tulajdonságait átróktörő állatrendszertanra, vagy egy számítógépes hálózat működésének hierarchikus leírására). Ezt a működést nem tudjuk igazán jól közelíteni, ha megmaradunk a hagyományos algoritmikus programépítési szemlélet mellett.

Mint látni fogjuk, egy tudásalapú rendszer célja nem input és output adatok közti leképezések végrehajtása, nem is adott feltételeknek elegendő tevő adatok kinyerése, hanem az, hogy adatokból (szak)ismeretekhez, szakértői szintű megállapításokhoz jusson el (például a beteg Morzsa kutya állapotát leíró adatokból megállapítsa, milyen kezelési módokat lenne célszerű alkalmazni a gyógyulás érdekében). Az output oldalán adat helyett információt jelenik meg: „Ezt és ezt tudtam kikövetkeztetni a (felhasználóval megbeszél) aktuális szituációból kiindulva”.

A tudásalapú technológia segítségével lehetővé válik személyes vagy kollektív tapasztalatok, heurisztikus jellegű ismeretek számítógépes ábrázolása és működtetése. Ilyen programok tervezése ezért nyilvánvalóan el kell, hogy térjen a hagyományos programok (azaz egy adekvát megoldást kereső algoritmus programjainak) tervezésétől. Kutató/felderítő jellegű (exploration-based) és modellközpontú tervezésre van itt szükség, amelynek jelenleg még mindig neuralgikus fázisa a szakspecifikus tudás megszerzése és rendszere-

zése (verbalizálás, elemzés, formalizálás stb.).

Képlet(es) játékok — nem képtelenség!

Az 1. ábrán látható képletre visszatérve, kizárólag a technológiai változások lényegének illusztrálása céljából, játszunk el a képletekben a program szövegtével. Az új igények figyelembevételével szervezzük át a programot a „strukturált programozás” korában jól hasznosított alapelveinek megfelelően. A bal oldalon két programrész van; képletebeli átszerkesztésünk során induljunk el először az algoritmusrészből, majd vegyük munkába az adatrészt:

1. játék

Daraboljuk fel értelmes, önálló részcsekkre (például eljárásokra) az algoritmusrészt. Foglalmazzuk át e részeket deklaratív stílusban, HA AKKOR szabályokkal ábrázolt „ismeretdarabkák” formájában (például: „HA a primer hűtőkörből a szekunder körbe a hőátadásának mértéke elégtelen, ÉS a víz áramlási sebessége alacsony, AKKOR a rendszer állapota miatt baleset várható.”). Megjegyzés: az adatok is ábrázolhatók így, például feltételrész vagy következményrész nélküli, elemi szabályokként.

Egészítsük ki ezt a készletet esetleg még további szabályokkal, majd helyezzük el ezeket egy tudásbázisnak nevezett rendszerkomponensbe, és működtetésüket bízzuk egy megfelelően kialakított végrehajtási mechanizmussal, egy ún. következtető gépre. Utóbbi egyrészt a következő feladatmegoldási lépésben végrehajtásra kerülő szabály kiválasztásával, másrészt a kiválasztott szabály végrehajtásával foglalkozik. Így egy ún. szabályalapú rendszerhez jutottunk.

2. játék

Az egymással kapcsolódó adatokat foglaljuk bele egy-egy adatraktúrába: frame-be/sémába/tudáskeretbe. (Kapcsolt adatokként, pontosabban objektum-tulajdonság-érték pármasként ábrázolható például a „Morzsa szőre fehér” állítás.) A tulajdonságok öröklődése szerint jelöljük be e struktúrák hierarchikus kapcsolatait. Egészítsük ki az így leírt világot úgy, hogy az minél teljesebb legyen, és a kapott struktúrák helyezzük el a tudásbázisban.

Ezután tördeljük fel és foglalmazzuk át az algoritmusrészt úgy, hogy az így

kapott darabkák (például eljárások — vagy akár az előbb látott szabályok) értelmesen hozzákapszolóhatók legyenek frame-eink bizonyos elemeire (például a hálózat egy pontján észlelt hiba azonnal hozzáz működésbe egy, a hiba hatásait továbbgyűrűztető eljárást, ún. démon).

Az így előállított „világéleírás” működésbe hozzáz kapcsoljunk be egy megfelelően kialakított öröklődési, esetleg egy démon mechanizmust is. Ha pedig úgy döntünk, hogy a frame-ek közötti információforgalmat magukra a frame-ekre bizzuk (ilyen értelemben a frame-eket önálló objektumoknak tekintjük), az egymás közti üzenetváltás elsőlegessége céljából kapcsoljunk be egy megfelelő üzenetváltó mechanizmust is. És így tovább. Mint látható, a következtető gép itt nem egy helyen van koncentráva, hanem „szét van osztva” (a 3. ábra csak elvi sémát). Az aktuális tudásbáziselem és a kívánt tevékenység kiválasztását a frame-ek adott hierarchikus kapcsolatai, valamint a frame-ek meghatározott pontjain elhelyezett információk (például az ezek hatására aktivizálódó démonok) vezérlik. A fenti mechanizmusok pedig végrehajtják a kiválasztott tudásbáziselemnek a szituációban megfelelő tevékenységet. Ilyképpen frame-alapú, illetve objektumalapú rendszerhez jutottunk.

Mindkét képletebeli játék lényege az volt, hogy kiemeltük a programból (annak procedurális kódjából) a tárgyerületi ismeretanyagot, és azt más szerkezetben ábrázolva behelyeztük a tudásbázis nevű, új rendszerkomponensbe. Továbbá, az 1. esetben az algoritmusrészből kiemeltünk egy végrehajtó-következtető gépet, míg a 2. esetben azt mintegy szétterítettük ástrukturált, hierarchikus elrendezésű világunkba, segítségül adva egy öröklődési, egy démon, egy üzenetforgalmazó biztosító stb. mechanizmust. Ástrukturált rendszerreink úgy működnek, hogy a pillanatnyi szituáció szerint tesznek aktívra egy szabályt, érvényesítenek valamely öröklődést, hajtanak végre egy démon, küldenek el vagy fogadnak egy üzenetet.

Maradjunk még továbbra is gondolati kísérletünkön. Amennyiben a fenti két átalakítást akképp végeztük, hogy a tudásbázisban, ahol csak lehetett, kódjuk helyett „neven neveztük” a dolgozat, megteremtettük a lehetőséget arra, hogy a tudásbázis elemeivel dolgozó mechanizmusok szimbólummanipulációt végezhessenek. Ez „olvashatóvá” teszi a tudásbázist, valamint segít a

felhasználói párbeszéd, valamint a magyarázatadás során a képernyőn megjelenő (természetes nyelvű) közlések futás közbeni generálásában is. Ezért van az, hogy az ilyen rendszereket szimbolikus programokként is emlegetjük.

A hagyományos program gondolati átszerkesztése egyben választ ad arra is, hogy hol keressük az adat, és hol az algoritmusrészt. Itt már nincs szétválasztva az adatelérés és adatkezelés az eljáráshívástól. Adat csak az input oldalon van; a rendszer output oldalán megjelenő üzenet már információ (példáknak: kezelési tanács a kutya tulajdonosának vagy az állatorvosának), amihez jelentést természetesen a végfelhasználó, az ember ad. Összegezve megállapíthatjuk, hogy:

— a „HOGYAN adjuk meg a függvényképezést realizáló algoritmust?” kérdésre választ adó algoritmikus programból egy,

— a „mely ismeretek megadásával tudjuk leírni a megoldandó feladatot?” hozzáféréssel készített, leíró, deklaratív szemléletű, szimbolikus programot kaptunk, amely a „MIT kell tudni?” szemléletet fejezi ki.

Mint már említettük, így olyan szerkezetű programhoz jutottunk, amelyről a tárolt ismeretanyagot már nemcsak egy feladat, hanem egy egész feladat-osztály esetén tudjuk mozgósítani (különösen, ha figyelembe vesszük azt, hogy átalakítás közben — a teljességre törekvően — ki is bővítettük a rendszerbe beépített ismeretanyagot).

Kreativitás és engineering

Az adatbázis-kezelő rendszerek megvalósításánál három fő elv érvényesült: nagyméretű adatbázisok nagyszámú felhasználó által egy időben történő használatának elve; adatfüggetlenség elve; a fizikailag különböző helyeken tárolt (ún. osztozt, elosztott, megosztott) adatokhoz való hozzáférés elve.

Közös az adatbázisokban és a tudásbázisokban, hogy — az emberek problémamegoldását segítő — számítógépekben információkat tárolnak. A jellegzetes különbség a tárolt információk típusában (adat — tudás/ismeret), valamint az általuk támogatott problémamegoldás eltérő jellegében van. Mély igazság van a következő, gyakran idézett képletben:

tudásbázis = kreatív adatbázis, amelyet ma sok kutató/fejlesztő „szó szerint” értelmez. Nevezetesen arról van szó, hogy a (hatékonyan működő)

adatbázis-kezelő rendszereket egy meghatározott irányban fejlesztik tovább: adatok mellett kifejezéseket, „ismeretforgácsokat” lehessen bennük tárolni, és azokat feladatmegoldásra mozgósítani (intelligens adatbázisok, objektumorientált adatbázisok — kibővített tudásalapú rendszerek).

Mint láttuk, a tudásalapú rendszerek a tárgyköri ismeretanyagot („tudást”) explicit módon ábrázolják; ennek főbb előnyei:

— természetes módon lehet új ismeretdarabkákat beépíteni, a régieket módosítani — lehetővé téve így az ún. inkrementális rendszerépítést (annak összes előnyével és veszélyével együtt);

— mind a rendszer pillanatnyi állapotáról, mind a problémamegoldás lépéseiről kézenfekvő a magyarázatadás, az indoklás: „WHAT (MIT) tartalmaz a tudásbázis (például: mely szabályok foglalkoznak a hőátadással)?”, „WHY (MIÉRT) tett fel a rendszer egy kérdést?”, „WHAT IF (MI LENNE, HA) ezt és ezt válaszolnánk?”, „HOW (HOGYAN) kezeltétek ki a rendszer az eredményt?”, „WHY NOT (MIÉRT NEM) ez és ez lett az eredmény?” stb.;

— a számítógépben ábrázolt tárgyerületi ismeretanyag és (az erről való tudás, azaz) a metatudás explicit ábrázolása előfeltétele az ismeretek automatikus „megszerzésének”, megfelelő tanulómechanizmusok kidolgozásának és beépítésének (amely alapkövetelmény a tudásalapú rendszerek következő generációjánál).

Ha tehát rendszereinket ellátjuk egy (természetes nyelvű, grafikus lehetőségekkel rendelkező stb.) felhasználóbarát felülettel, valamint (az embert utánzó) megfelelő magyarázatadási képességgel (legalább a WHAT, WHY és HOW lehetőségekkel), akkor egy újszerű programszervezési elvek szerint felépített, ún. tudásalapú rendszerhez jutottunk. Egy ilyen tudásalapú rendszer működését a felhasználó jellemzően nem belépési pont (vagy valamely lekérdőző nyelven megfogalmazott kérdés cél) megadásával indítja, hanem — mint intelligens beszélgető partnerrel — „konzultál” vele az aktuálisan megoldandó problémáról (adatot közöl vele, a rendszer felkérésére dönt egy adott kérdésben, magyarázatokat, indoklásokat kér a rendszertől stb.).

Specialisták és „konziliumok”

Két új szereplő lép előtérbe: a tudásalapú rendszerépítésben jártas tudásmémők (ismerettechnológus) és a szak-

ismereteit átadni képes tárgyköri szakértő. A végfelhasználó is más, mint eddig. Nem „kezelet” a programot, hanem tanácsadó partnerként „megbeszéli vele” a problémát, a rendszer által adott indoklások figyelembevételével átgondolja a rendszer javaslatát, majd dönt (a döntés felelőssége az emberé!). Egy tudásalapú rendszer tehát intelligens információfeldolgozó rendszerek tekinthető. Bár a valóság ennél bonyolultabb, már az eddigiekből is látszik, hogy a tudásalapú technológia forradalmasította nemcsak a rendszerépítést, hanem az ember-gép kapcsolatot is.

Tudásalapú rendszerek kifejlesztésére és futtatására ma már sok keretrendszer, shell található a piacon; ezek segítenek az új technológia elterjesztésében, valamint egyszerű rendszerátvitelhez, prototípusok elkészítésében. Az első shell megjelenése után egy ún. technológiavezérelt, prototípus-centrikus rendszerépítési hozzáférés volt jellemző: adott shellhez, eszközkészlethez keresni megfelelő alkalmazásokat (rossz esetben: minden alkalmazást egy adott shell felhasználásával megvalósítani). Ebben a szakaszban sok prototípus készült világszerte, de igazán csak 20 körüli volt használatban (1988 végén a piac befagyott látszott).

Jelenleg egy egészségesebb, ún. „feladatvezérelt”, modellvezérelt rendszerépítési szemléletmód kezd egyre jobban terjedni. Ennek lényege a következő: ismerjük meg előbb az adott feladat természetét, jellegét, majd ezután keressük csak meg a („hozzá legjobban illő”) megvalósítási eszközközt.

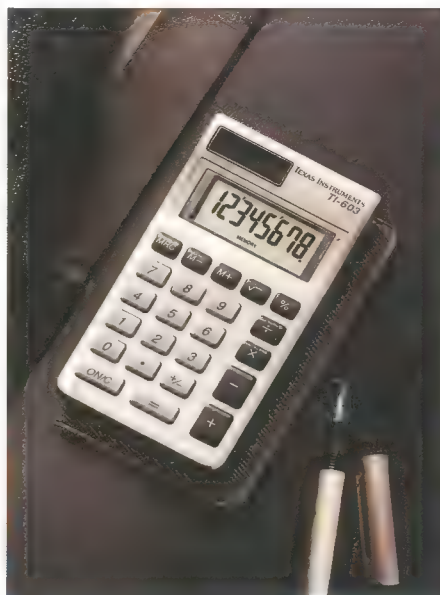
Elemi igény itt is a korábbi technológiákkal való „együttélés”: szükség van például számszaki számítások hagyományos nyelvű eljárásoknyitvainak, grafikus eljárásokéinak stb. futás közbeni felkeresésére, nagy adatbázisokban tárolt adatokhoz való hatékony hozzáférésre, egymással együttműködő tudásalapú rendszerek egyazon számítógépes hálózatra kapcsolására stb. Mindezek megvalósításához intelligens komplex rendszerek szükségesek; ez pedig elemi igényként veti fel a hagyományos programozási iskolák, valamint az adatbázis-technológiák és a (még „éretlen”) tudásalapú technológia integrálásának szükségességét.

Ebbe a vonulata a hazai szoftverfejlesztők és szakértők elég hamar bekapcsolódtak. Az Alaplap következő számában néhány ígéretes munkáról olvashatnak a már megvalósult, működő szakértő rendszerek iránt érdeklődők.

Sztáné Tóth Edit



**SZÁMOLÓGÉPEK ÉS ADATBANKOK
TELJES VÁLASZTÉKA
A BÜROTECH KFT-TŐL
TELEFON/FAX: (27) 58-308**



Tudjuk, hogy nem csak ketten vagyunk a világon:

MI ÉS A *PACKARD BELL*

Egy biztos: EZ EGY JÓ CSAPAT!



DISTRIBUTOR

**SZINVA NET
SZÁMÍTASTECHNIKAI
SZOLGÁLTATÓ
ÉS KÉRESKEDelmi KFT.**

3525 Miskolc, Völgyda utca 3.
Levelezési cím: 3545 Miskolc, Postaórák 448
Telefon: (46) 355-895
Kereskedelmi részleg: Telefon: (46) 346-634
Szervizrészleg: Telefon: (46) 340-841
Budapesti Iroda:
1149 Budapest, Bosnyák tér 5.
Telefon/Fax: (1) 252-0545

SZOLGÁLTATÁSOK

Minden géphez díjmentes
- jogtisztaság MS-DOS 5.0
- LOTUS WORKS integrált programcsomag
- DOSABC magyar oktatási program
- EKSZER magyar nyelvű szövegszerkesztő
Hálózati lehetőségek:
- TOKEN RING
- ETHERNET
Jogtisztaság NOVELL



3525 Miskolc, Déryné u. 18.
Tel./Fax: (46) 347-888
Kazinczy u. 19. Tel.: (46) 349-619
Eger, Csiky S. u. 17. Tel./Fax: (01) 06 106
Levelezési cím: 3501 Miskolc, Pf.: 398

Rádiófrekvenciás
azonosítás
az állattenyésztésben
és az iparban.

TIRIS

**A
MISKOLCI IPARI VÁSÁRON
a COMPUTER PRAXIS Kft
kiállítási területén.**

1992. augusztus 14-től 23-ig.



„THE MACRO” számítógépek,
NOTEBOOK computerok,
STAR nyomtatók és kiegészítők,
3M mágneses adathordozók,
GENIUS mouse-ok, scannerek
UPS szünetmentes tápegységek,
CADDY grafikus tervező rendszerek,
ÜGYVITELI és GYÁRI szoftverek.

Látogasson el hozzánk!



Macroda Kft. mintabolt:
1123 Bp., Alkotás u. 21.
Tel.: 201-4603
Tel/Fax: 156-4802

MEGFEJTETTÜK A TITKOT!

A jó üzlet titka:

- ① széles áruválaszték, ② gyors és pontos kiszolgálás, ③ hozzáértő szaktanácsadás.

A MACRODA Kft. mintaboltjában mindez megtalálható.

MACRODA - AZ IDOMÍTOTT SZÁMÍTÁSTECHNIKA

INFORMÁCIÓKÉRÉS: 18 ▲



Pc-Comp
Számítástechnikai
Szolgáltató és
Kereskedelmi Kft.

PC-k és perifériák forgalmazása garanciával.
Hálózattelepítés, installálás -Authorized Novell Dealer
IBM és kompatibilis PC-k és perifériák (monitorok,
tápegységek, nyomtatók) javítása.
Átalánydíjas karbantartási szerződések
kedvezményekkel! Szaktanácsadás.

**”Ami elromolhat, az el is romlik”-
de Mi megjavítjuk!**

1078 Budapest Murányi u. 41.
Telefon / Fax: 142-7202



BONUS
3% értékhelyettesítő
vagy szolgáltatás igénybe-
vétele esetén.

INFORMÁCIÓKÉRÉS: 19 ▲

PADSor I.

Vége a mellőzöttségnek

„Ingyen PADS” akció keretében a komplett PADS (Personal Automatic Design System) alaprendszer limitált változatához juthattak hozzá az érdeklődők. A 6000 elemes alkatrészkönyvtárt is tartalmazó szoftver vizsgálatá során — amelyről két részben számolunk be — kiderült, hogy a „kis rendszer” tényleg tudja mindazt, amit beharangoztak róla: igazán felhasználóbarát környezetben 60-70 alkatrészből felépített áramkörök gyártásra készre(!) tervezhetők vele.

A könnyen kezelhető, moduláris, PC bázisú professzionális PADS rendszernek először a layout-tervező (PADS-PCB), majd a sémamodifikátor (PADS-Logic) változata készült el. A fejlesztés újabb lépcsőfokát a PADS-2000 program jelentette, amely — a konkurens programcsomagokat megelőző — PC-s környezetben nyújtja a munkaállomás alapú tervezőrendszerek legfontosabb szolgáltatásait. Ugyanakkor napjaink igényeinek megfelelően megszületett a rendszer Unix alapú, Sun Sparcstation munkaállomásokon futó változata is, ez a PADS-2000/UX.

A PADS megjelenése előtt az analóg áramkörök konstruktőrei úgy érezték, hogy „mellőzöttek”, hiszen a tervezőprogramok inkább a digitális áramkörök technológiájának kedveztek. A PADS fejlesztői szakítottak ezzel a filozófiával. Így a szoftver számos olyan szolgáltatást tartalmaz, amely analóg és nagyfrekvenciás áramkörök tervezésére is alkalmas. Nem véletlen tehát, hogy a PADS rövid idő alatt közzismert és közkedvelt lett a nyáktervezők körében.

Redundanciamentes

A PADS rendszer része az alkatrész-könyvtár, amely jól felépített szerkezetével szükségtelenné teszi a nagy mennyiségű adat és grafikus elem redundáns tárolását. Az alkatrészkönyvtár négy része az adatlap-, a szimbólum-, a PCB formakönyvtár, valamint a grafikai elemek könyvtára.

Az adatlapkönyvtár tartalmazza az elektromos elemek részletes paraméte-

reit: az elem nevét, logikai családját, a variánsok (az azonos funkciójú, kivezetésű, elrendezésű elemek) listáját, valamint az elemen belüli azonos funkciók (tipikusan kapuk) számát, felcserélhetőségüket. Itt utalunk az elemhez tartozó rajzsimbólumra, a tokozásra, valamint azok alternatív megfelelőire. Rajzsimbólum esetén ez lehet a logikai kapu De Morgan-megfelelője, míg a tokozásnál a ténylegesen létező összes típus szerepel. Konkrét grafikus definíciókat azonban az adatlapkönyvtár nem tartalmaz.

A szimbólumkönyvtárban azok a grafikai jelölések szerepelnek, amelyek a kapcsolási rajz lapjain találhatók meg. Ennek elemeit grafikus támogatással a felhasználó is bővítheti, amelyhez egy

általános célú grafikai könyvtár elemei közül válogathat.

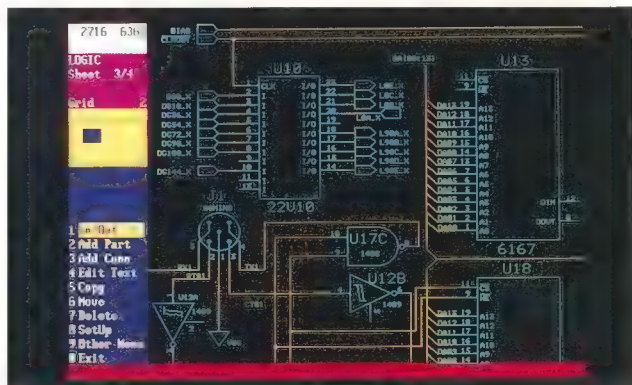
A PCB-formakönyvtárban található meg az egyes tokozási típusok definíciói, a tok körvonala, a kivezetések helyzete, mérete, továbbá a forrsemek alakja. A formakönyvtár természetesen tovább bővíthető.

A grafikai elemek könyvtára olyan szimbólumokból áll, amelyeknek nincs elektromos tartalmuk. A szimbólumok a kapcsolási rajzon és a nyomtatott áramköri panelen egyaránt szerepelhetnek. A felhasználó kialakíthatja például cége emblémáját, amelyet a továbbiakban minden rajzon és terméken elhelyezhet.

A nagy helyigényű grafikus szimbólumok csak egy példányban szerepelnek az alkatrészkönyvtárban, így a tárolás „takarékos”. A redundanciamentes tárolási eljárásnak köszönhetően a jelenleg 7400 különböző elemet tartalmazó alkatrészkönyvtár valójában 15-16 ezer alkatrészt jelent — mindössze 3 megabájnyi lemezterületen.

Sémákban gondolkodik

A PADS-Logic modul a rendszer sémamodifikátor. Segítségével rendkívül egyszerűen szerkeszthetők meg a kapcsolási rajzok. A program 16 és 32 bites változatából 286-os gépen a 16 bites, 386/486-os gépen pedig a 32 bites



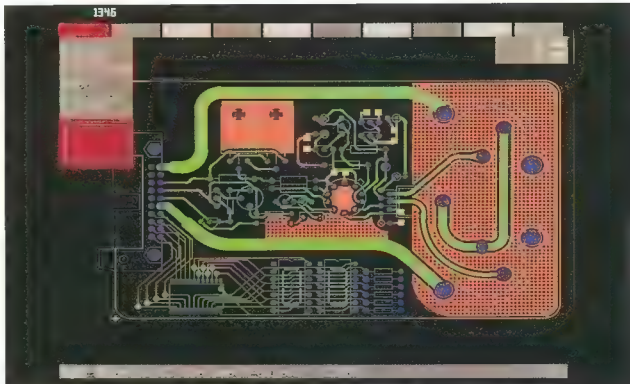
változatot installálják. Használata EGA vagy VGA videoadapter, egeret, legalább 2-4 Mb-át memóriát igényel. Alapában igaz, hogy a 32 bites verzió 4-5-ször több alkatrészt, összeköttetést kezel, mint a 16 bites változat.

A PADS-Logic program kezelői felülete azonos a rendszer többi moduljával. Kidolgozásakor alapelv volt az egér és a billentyűzet együttes használata, mivel így lényegesen gyorsabban érhető el az egyes funkciók. A 16/32 bites változatban a képernyő bal oldalán/tetején látható a 10 elemből álló menürendszer, amely a billentyűzet 10 funkcióbillentyűjének felel meg. A menüt úgy alakították ki, hogy egy funkciót több helyről is elérhetünk ugyanazzal a billentyűvel. Például az F10 mindig kilépést jelent, a képernyő alsó sorában pedig szöveges információk vihetők be. Ahol alkatrész- vagy fájlnevet kell megadni, ott a * jel hatására egy listát kapunk a választható elemekről.

Nagyan megkönnyíti a program használatát, hogy a grafikus szimbólumok szerkeszthetők. Például ha meg akarjuk változtatni valamelyik szimbólumot, vagy egy újat akarunk létrehozni, ezt a Logic programból való kilépés nélkül tehetjük meg.

Figyelemre méltó a képernyőn látható szálkereszt helyzetérzékeny kiválasztása. Így a mozgás vagy törlés parancs használatakor nem kell külön megválasztani, hogy milyen típusú objektumot akarjuk a műveletet elvégezni, elegendő ezt a szálkeresztet kiválasztani.

A felhasználó által definiálható makrók a többször ismétlődő műveletek egyszerű és gyors végrehajtását teszik lehetővé. Az ún. ideiglenes makró csak



a program egyszeri futása alatt használható, míg a különböző billentyűkhöz rendelhető állandó makrók elmenthetők, bármikor felhasználhatók, utólag megváltoztathatók.

Mivel a korszerű, nagy integráltságú elektronikus kártyák kapcsolási rajzai rendszerint olyan mennyiségű alkatrész tartalmaznak, hogy azok egyetlen elfogadható méretű rajzlapon sem jeleníthetők meg, a tervet rajzlapokra kell bontanunk. Ez a felbontás lehet hierarchikus vagy mellérendelt. Az előbbi esetben a fő (a hierarchia csúcsán álló) rajzlapon elhelyezett modulok egy-egy önálló lapot jelentenek, míg az utóbbinál az egyes lapokat egymás mellett kell elképzelnünk. A program nem korlátozza a hierarchikus felépítés szintjeinek számát.

A PADS-Logic egyik nagyon fontos jellemzője — ami korábban csak a munkaállomásokon működő rendsze-

rek sajátja volt — a kártyaorientáltság. A hasonló sémaeditorok rendszerint laporientált működésűek. A felhasználás szempontjából fontos megkülönböztetni a két elvet. A laporientált sémaeditorok logikai egysége a rajzlap, így egyidejűleg csak az aktuális lapon levő alkatrészek és kapcsolatok töltődnek a számítógép memóriájába. A kártyaorientált PADS-Logic viszont együtt kezeli a teljes terv adatbázisát. Így lehet például egy jelet vagy egy áramkört elemet — bármelyik lapon is legyen — könnyedén megtalálni.

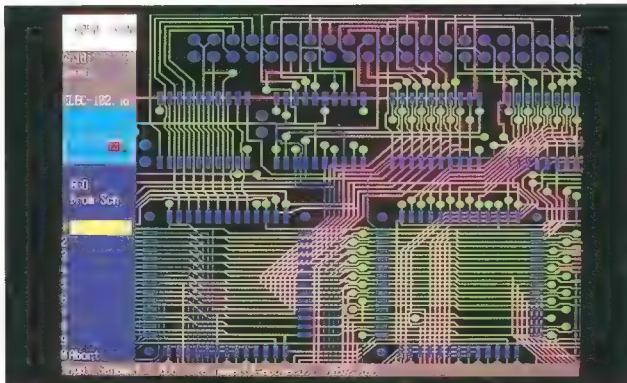
Ugyancsak nem kell kilépni a szerkesztőprogramból ahhoz, hogy elvégezzük a terv logikai helyességének ellenőrzését és az automatikus alkatrész-sorszámozást.

A PADS-Logic menüjébe beillesztették azokat a programmodulokat is, amelyek a kapcsolási rajzból alkatrész- és összeköttetési listát állítanak elő. Az alkatrészlista tetszőlegesen formázható, ízlésünknek megfelelő címeket, tételeket definiálhatunk — mindezt magyarul is.

A teljes tervadatbázis (vagy annak bizonyos részletei) kimenthető vagy betölthető ASCII-formátumban is. Ez megkönnyíti az illeszkedést olyan rendszerekhez, amelyekhez ezt a PADS nem biztosítja. Ugyanakkor az ismertebb tervezőrendszerekkel „közvetlen a kapcsolatot”.

Manuális és automatikus

A kapcsolási rajz hibátlan elkészítése és ellenőrzése után hozzáférhetünk a tényleges nyomtatott áramkörti lap tervezéséhez. Ebben segít a PADS-PCB modul, amely olyan interaktív szerkesztőprogram, amelyet felkészítettek a kü-



lön megvásárolható automatikus alkatrész-elhelyező, huzalozó- és a gyártási kimenetet előállító modulok integrálására. A PADS-PCB — ezen modulok nélkül — az alkatrészek manuális elhelyezését és huzalozását végzi el. Kezelői felülete megegyezik a Logic modulal.

A PADS-PCB kapacitása legfeljebb 250 ekvivalens IC méretű kártyák kezelését teszi lehetővé. A PCB modul fontos feladata a kártya fizikai méreteinek definiálása. Segítségével a pontos kártyakontúr inchben vagy metrikus egységben is létrehozható.

Az alkatrészek elhelyezéséhez először az ún. pozicionálási mátrixot kell létrehozni. A mátrix pontjai szolgáltatják az alapot az alkatrészek automatikus elhelyezéséhez. Az alkatrészek egy csoportja azonban igényli a kézi elhelyezést. Ilyenek a kártyán a rögzített helyű csatlakozók, élcsatlakozók, kapcsolók, potenciométerek, LED-ek, és minden más olyan elem, amelynek igazodnia kell valamilyen külső geometriához. Ezek az alkatrészek a PCB modul segítségével pozicionálhatók végleges helyükre. Bizonyos analóg vagy nagyfrekvenciás alkalmazások is szükséges-

sé tehetnek olyan különleges alkatrész-elrendezést, amelyet a tervező csak manuálisan valósíthat meg.

A PCB modul hatékony segédeszközökkel támogatja az alkatrészek kézi elhelyezését. Különböző csoportosításban automatikusan adagolja az elemeket, mozgítás közben folyamatosan mutatja az új elem kivezetéseinek leg-rövidebb bekötését (rubber banding). Az alkatrészek forgathatók, tükrözhetők, kicserélhetők más tokozású változatra. A kötött helyű alkatrészek kézi elhelyezése után elindíthatjuk a PADS-Place programot, amely a hátralevő alkatrészek automatikus elhelyezését végzi el. Amennyiben valamilyen ok miatt elégedetlenek vagyunk az eredménnyel, utólag természetesen kívánságunknak megfelelően módosíthatjuk is az elrendezést.

Az alkatrészek elhelyezése után a PCB modul kézi huzalozásra ad lehetőséget, ami jól jön a kritikus vonalvezetési jelek útvonalának kialakítása során. A jelek eközben tetszőleges szögben vezethetők, de csak merőleges vagy 45 fokos vonalvezetés is előírható.

A kézi huzalozás közben folyamatos információt kapunk az aktuális jekről.

Meglévő jelvezeteket vagy annak egy részletét is szerkeszthetjük, mialatt hozzáadhatunk vagy törölhetünk töréspontot a jel útvonalából, átérhetünk más rétegre, illetve vonalvastagságot változtathatunk.

Átjárható

A Logic modulall kialakított kétirányú kapcsolat révén a kapcsolási rajzon végzett utólagos módosítások a kártyatervre átvihetők, s a kártyatervezés közbeni változtatások (alkatrész-átnevezés, kapu- és kivezetéscserék) is visszajelölhetők a kapcsolási rajzra. A PADS rendszerben ECO (Engineering Change Order) kapcsolatnak nevezik ezt a lehetőséget, amely minden esetben biztosítja a kapcsolási rajz és a kártyaterv logikai összhangját.

Az igazán nagy méretű kártyák tervezői a PADS-Large modulall egyfajta teljesítményfokozást érhetnek el, segítségével a PADS-PCB határadatai mintegy másfélszeresre növelhetők. Működtetéséhez a PCB modul memória-igényén túl további 1 Mb-ot EMS memóriára van szükség.

Lóth Tamás

A készüléket alapképzésben felkészítésként a bővítés összes lehetőségére.

Hálózati teljesítménye, csatlakozópontjai, I/O kártyái, rendszerelemei, speciális zavarmentesítő árnyékolása egyaránt a kiterjesztést szolgálják.

Kérjen részletes információt az

INTEL Xpress
cserélhető kártyás
formatervezett készülékről a forgalmazótól:

**MOST BÁTTRAN HIHET A JÓSLATNAK:
AZ XPRESS SZÁMÍTÓGÉP
A BIZTOS JÖVŐ!**



a **COMPMARK** Kft.-től.



COMP MARK

Számtástechnikai és Kereskedelmi Kft.

☐ 1138 Budapest XIII., Párkány u. 20.

☎ 1731-272, 1731-358 • Fax: 1731-272

Akik szeretik az egyszerűt, figyeljenek!

Csaknem Basic

Az #539 számú SolarSoft lemezen található Asic egy Basic programozási nyelv IBM PC-re.

(Az Asic elnevezés eredete: „It's Almost Basic”, ami magyarul csaknem Basic-nek fordítható.) Integrált editor tartalmaz, ebből szerkeszthetjük, fordíthatjuk, tesztelhetjük programjainkat, de természetesen saját kedvenc editorunkkal is dolgozhatunk, DOS-ból is fordíthatunk.

Az Asic a Basica és a GWBasic nyelvek alapján készült, 80-nál több Basic utasítást támogat, integer és string változókat, stringsorozatokat kezel. Fordítása gyors, az általa generált kód tömör. Az Asic utasítások csaknem mindegyike megtalálható a Basica és a GWBasic nyelvekben is.

Használatához 305-405 kb-át szabad memória szükséges. Maga az Asic fordító 305 kb-át memóriát igényel, az editornak elég 200 kb-át nál valamivel kevesebb. 405 kb-át akkor szükséges, ha az integrált editorból akarunk fordítani és hibát keresni.

Az Asic nyelv szabályainak megfelelő programok mindenki számára ismerősek lesznek, aki látott már Basic programot. Az első szembetűnő különbség az, hogy itt nem sorszámozunk. Nézzünk egy rövid programrészletet:

GWBasic/Basica

```
10 I=I+1
20 PRINT I
30 IF I<100 THEN 10
Asic
ADDONE: I=I+1
PRINT I
IF I<100 THEN ADDONE:
```

Mindkét program az 1 és 100 közötti számokat jeleníti meg a képernyőn. Az Asic azonban beszélő címkéket használ a Basic sorszámkok helyett. Ezek a címkék akár 80 karakter hosszúak is lehetnek. Vegyük észre azt is, hogy az Asic PRINT és IF utasításait nem kötelező beljebb kezdeni, ez csak a program olvashatóságát könnyíti meg! Írhamánál összesen ennyit:

```
Asic
ADDONE:I=I+1
PRINT I IF I<100 THEN ADDONE:
A hatás ugyanaz lenne. A változónevek is lehetnének hosszabbak, egészen
```

80 karakteresig. Ugyanez vonatkozik a stringkonstansokra is.

Az Asic programok beville a lehető legegyszerűbb dolog. Soronként begépeljük, az egyes sorokat ENTER-rel zárjuk. Integrált környezetben a képernyő 2-24 soraiba írhatunk. A felső sor mindig a menü, a legalsó pedig a státusz-sor.

Induláskor négy menüből választhatunk: File, Edit, Compile és Run. Nevük önmagáért beszél. Run választás esetén nemcsak futathatjuk, hanem tesztelhetjük is (az Asic debuggerrel) programjainkat. Az alsó menüsínteken egyes műveleteket kivettek a menüszervezetből, ezeket funkciók billentyűkhöz rendelték.

Online help segíti a felhasználókat. Asic-kulcsszavakat, ezek szintaxisát, hatásukat; az egyes menüpontokkal kihasználható tevékenységeket jeleníti meg a képernyőn.

A File menü állományok megnyitására, módosított tartalom mentésére

(esetleg más néven) szolgál. Ebből a menüpontból tudunk visszalépni DOS szintre.

Edit menüben a szövegszerkesztésnél megszokott műveleteket végezhetjük: blokk-kijelölés, törlés, áthelyezés, másolás, karaktersorozat keresése, helyettesítés, ismételt keresés. Ami már több ennél: itt indíthatjuk a fordítást, itt állíthatjuk be a fordító kapcsolóit, köztük a nyomkövetést is. A fordítás megkezdése előtt a rendszer természetesen elmenti az aktuális forráskódot.

A státusz-sor editor módban a megnyitott állomány neve mellett "*" gal jelzi azt, hogy már módosítottunk valamit, de nem mentettük el az új tartalmat. Megjeleníti az aktuális sor állományon belüli sorszámtól és az ablakon belüli sor-oszlop kurzorpozíciót is. Editor módban a billentyűzet a megszokott módon viselkedik, az esetleges eltéréseket a kézikönyv tartalmazza.

Run menüből indíthatjuk a sikeresen lefordított, már hibátlanunk tűnő programokat, ki-be kapcsolhatjuk a nyomkövetést, az utasításokonkénti végrehajtás-felfüggesztést, módosíthatjuk az egyes változók értékeit, megjeleníthetjük a kijelölt változók aktuális értékeit, megszakíthatjuk a programfutást, a töréspontoknál megjeleníthetjük a program által írt képernyőt ...

A debugger segítségével futás közben vizsgálhatjuk programunk működésének helyességét. Akár utasításokonként hajjuk végre a programot, akár az általunk beiktatott töréspontoknál felfüggesztve, tetszőleges változók aktuális értékeit nézhetjük meg, módosíthatjuk is ezeket. A nyomkövetéssel kapcsolatos parancsokat vagy a „Run” menüpontban adjuk meg, vagy az e célra szolgáló funkciók billentyűkkel (menün kívüli). A nyomkövetést oktató segédállomány is megtalálható a lemezen. Tíz

Fordító

A fordító opciói:

C (CONTINUE) üzenetek kiadása után nem áll le az Asic, hanem folytatja a fordítást;
L (LIST ALL) fordítások minden egyes lefordított sor listáz (alapértelmezésben csak a hibás sorokat listázza);
P (PRINTLIST) a képernyőn megjelenő üzeneteket ki is nyomtatja;
D (DISKBACKLIST) a képernyőn megjelenő üzeneteket lemezes állományba is tárolja (ez gyorsabb, mint az előző opció);
S (SYM/TAB) lemezes állományt készít a szimbólumokról (az összes változó, címke, belső fordítási címek);
E (EMATH) bekapcsolja az „Extended Math” opciót (ekkor tudunk long integer típusú

változókkal, azaz +2,147,483,647 és -2,147,483,647 közötti értékekkel dolgozni);
X — DEBUG — nyomkövető kódot generál.

A fordító korlátai:

A forrásprogram maximális hossza integrált környezetben 809 sor, önállóan azonban nincs korlátja. A tárgyidő maximális hossza 64 kb-át lehet. A forrásprogram egy-egy sora 128 bájttal kezdődhet, 80 bájttal az integrált környezetben.

Ciklusok 25 szintig akatulyázhatók egymásba. Maga a fordító egyenesen, Borland Turbo C 2.0-ban írt. A forrástulajdonosok közvetlenül gépi kódú alakjait. Teljesen a memóriában dolgozik, így érheti el a legnagyobb sebességet. Lemezerületet egyáltalán nem igényel.

számot rendez „buborék” módszerrel — ezen a példán jól be lehet mutatni a debugger használatát.

Asic-ben nincs EOF, sem ON ERROR. Ezek helyett egy speciális rendszerváltozó (neve ERROR) értékét állítja be akkor, ha állománykezeléskor (OPEN, INPUT#, PRINT#, CLOSE ... utasítások végrehajtásakor) hibát észlelt. Az Asic állományok 2/4 bájtis integer formátumokból állnak (normál/long integer), ellentétben a Basica és GWBasic stringformátumával.

Itt nem olyan szigorúak az frásjelek alkalmazási szabályai. Megengedi ugyan használatukat, de nem követeli meg.

A = ABS (B(1)) helyett nyugodtan írhatjuk:

A = ABS B 1
vagy akár:
A = ABS B 1,,,,,))))))

A FOR/NEXT ciklusokat legalább egyszer végrehajtja, ez sem így történik Basica/GWBasic-ben. Kerekítéskor (RND) 0 — 32 767 közötti értéket kapunk normál integerekénél, 0 — 2.147.483.647 közötti long integerekénél, míg Basica/GWBasic esetén ez 0 és 1 közé esik.

A soros portokat támogató OPENCOM / COMSTAT / SEND / RECEIVE utasítások GWBasic/Basica OPEN/INPUT#/PRINT# -nek felelnek meg. Megengedi a rendszer az idő és dátum lekérdezését, de módosítását már nem.

Új utasítások Asic-ben:

CODE COMMANDS

Nyomkövetési tippek

1. Amikor már helyesnek tartjuk programunk működését, tehát nincs tovább szükségünk nyomkövetésre, fordítsuk le újra a DEBUGGING CODE opció nélkül (OFF). Így nem növeljük feleslegesen sem a futás idejét, sem a kód méretét, mivel gyorsabb és rövidebb programokkal dolgozunk.
2. Figyeljünk arra, hogy míg az integrált környezetben a DEBUGGING CODE ON az alapértelmezés, külön fordítva OFF.
3. Állományok megnyitásokkor és fordításakor az addig beiktatott függésjelesi pontok automatikusan törölődnek.
4. Ha futó programban változtatunk meg egy forrást (újrafordítás nélkül), akkor a forrásutalítás nem lesz azonos a végrehajtott utasítással. Ezt ne csináljuk soha! Szakítsuk meg a programfutást, és fordítsuk újra!
5. A függésjelesi pontok helyét a rendszer az editor/puffer elejéről számítja. Kavarodást okozhat a fordítás/beírás, mert a rendszer továbbra is ezzel a relatív sorozámmal dolgozik.
6. Az Asic az INT 3 interruptvektort használja nyomkövetésre. Ennek eredeti értékét akkor állítja vissza, amikor kilépünk az Asic-ből.
7. Forrásszintű nyomkövetéshez a DEBUGGING CODE opciót meg kell adnunk fordításakor.
8. Ha futó programunk futását előlőrl akarjuk kezdeni, először válasszuk a Run menü „Abort

your program” opcióját, utána pedig a „Run your program” opciót!

9. „Abort” opció választásakor a programhoz tartozó adatállományok lezárása nem lesz korrek. (Az adatpufferek tartalma nem kerül vissza a lemeze.)

10. Array változók módosításakor ügyeljünk az indexhatárok betartására! Ezt ugyanis nem ellenőrzi az Asic.

11. Zavaró lehet, hogy a fordító max. 80 karakteres változókkal dolgozik, míg a nyomkövető csak 30 karakterekkel. Erre is nekünk kell ügyelnünk.

12. FOR — NEXT ciklus FOR sorába tett töréspontnál csak egyszer fog megállni a program. Ciklus végén már csak a FOR-t követő utasításra ugrik vissza!

13. A CODE utasítás sorában nem lehet töréspont. Ez logikus, ugyanis egy gépi kódú utasítás több CODE utasítást is tartalmazhat.

A nyomkövetés korlátai: max. 20 töréspont, egyidejűleg 10 szimbólum aktuális értékét figyelhetjük, az egyes szimbólumok maximum 30-30 karakter hosszúak lehetnek, a forrásutalítások száma legfeljebb 809 lehet, a szimbólumok maximális számát nem a nyomkövető, hanem a fordító korlátozza.

INT86
FIND
LTRIM\$
RTRIM
ZBIT

FILEPOS
LCASE\$
MOD
UCASE\$
ZMODE

Ezeknek nincs megfelelőjük sem GWBasic, sem Basica nyelven.

A 124 oldalas kinyomtatható angol nyelvű kézikönyv mellett két Asic nyelven írt demoprogramot is találunk. az egyik az állományok kezelésére tar-

talmaz példákat, a másik pedig a valószínűleg közismert cellakezelés/eltérő folyamatának forráskódját. Itt 20x80 karakteres területen minden olyan cella elhal, amelynek kettőnél kevesebb szomszédja van (túlszűzítés), vagy háromnál több (túlnépesedés). Azok maradnak életben, amelyeknek 2 vagy 3 szomszédjuk él. Az üres helyeken új cellák keletkeznek, amikor 3 élő szomszédjuk lesz.

Sándor András

Mi muzsikos lelkek...

A nyári SolarSoft újdonságok között háromat is találunk, amely a számítógépes zene iránt érdeklődők számára kellemes meglepetést jelent: az #516 lemez a rég várt Blaster Master 4.8, az #517 a SoundFX-Blast, az #518 pedig az Audiostar programot tartalmazza, Ezúttal az első kettőt ismertetjük.

Azoknak készült a Blaster Master, akik a maximumot szeretnék kizozni PC-jük hangkártájából. A SoundBlaster és más hangkárták egyik legérdekesebb tulajdonsága, hogy képesek a hangokat felvenni és visszajátszani is. A Blaster Master alapértelmezésben SoundBlas-

ter típusú .VOC kiterjesztésű állományokkal dolgozik. A regisztrálatlan shareware verzióval csak 25 másodperces hangállományok kezelhetők, a regisztrált verzióval nincs ilyen korlát. Külső hangállományokat is kezelhetünk a rendszerrel: .WAV, .NTI 8SV és

.SND formátumokat. A .WAV állományok Microsoft Multi-Media formátumúak, ezeket többek közt a Pro Audio Spectrum és az ATI Stereo F/X hangkárták támogatják. Az .SND kiterjesztésű hangállományok csak adatokat tartalmaznak, nincs header információ bennük. Az .NTI állományok olyan Amiga mintákat tartalmaznak, amelyeket a Tetra Compositor kezel. A .VOC típusú állományokat mindhárom említett formátumban tárolhatjuk. A .8SV kiterjesztésű az Amiga IFF hangállományok alapátípusa. Fel tudjuk ugyan dolgozni az Amiga IFF hangállományokat, de pillanatnyilag még nem menthetjük el saját mintáinkat ilyen típusú állományba.

Maga a program egy igen gyors EGA/VGA grafikus editor, amelynek segítségével kívánságainknak megfele-

lően alakíthatjuk hangállományainkat. Eltávolíthatjuk például a szükségtelen mintákat az elejéről, végéről; kiválaszthatunk részeket, ezekből új állományt képezhetünk. A kivágási pontok pontos kiválasztásakor nagy segítséget jelent, hogy a teljes minta grafikusan jelenik meg a képernyőn, másodpercenkénti osztásban. A feldolgozáshoz a legkisebb pontok kiválasztása a 0 átmenetek alapján történhet. A képernyő színei funkcióbillentyűkkel (F1—F5) változtathatók. Visszajátszáskor pulzáló jel mutatja, hogy hol tartunk a dallamban. A digitális feldolgozási folyamatok aktivizálása egérrel történik, az esetek többségében az Alt billentyű és a funkció első betűje is ugyanezt az eredményt adja.

Szerkesztéskor blokkműveleteket is végezhetünk. A kiválasztott blokkot megismételhetjük (maximum tízszer), csendbájtokkal tölthetjük fel, megfordíthatjuk — így visszafelé fogja lejátszani —, visszahangosíthatjuk.

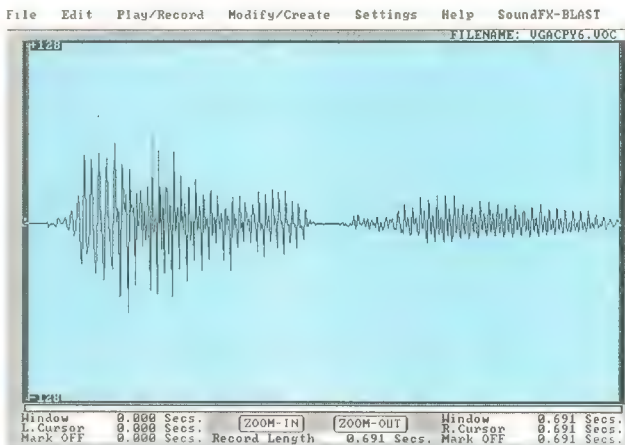
Érdekes kipróbálnunk a 12 000 Hz-en felvett .VOC állományok 8000 Hz-es tárolását. Sokkal jobb hangzást érünk el így, mintha már eredetileg is az alacsonyabb frekvencián vettük volna fel. Úgy lesznek rövidebbek az állományok, hogy a hangzásban nem érződik a tömörítés miatti veszteség. A hangzás tisztasága megmarad akkor is, ha a MAC-en eredetileg 22 000 Hz-cel felvett mintákat 12 000 Hz-en játszuk le.

Felvételkor érdemes a kisebb hangerőt választanunk, és csak később átállítani nagyobb hangerőre. A SoundBlasternek ugyanis van egy olyan tulajdonsága, hogy nagyobb hangerő esetén erősebb a zavarás, de természetesen ekkor a PLAY, RECORD és a SCOPE művelet értéketlenek.

Speciális effektusokat érhetünk el két állomány keverésekor a hangerősségek változtatásával.

A Blaster Master hangkártya nélkül is működik, de természetesen ekkor a PLAY, RECORD és a SCOPE művelet értéketlenek.

Szükséges konfiguráció: EGA/VGA grafikus kártya, egér, merevlemez, lehetőleg gyors PC. Mivel szerzője lebegőpontos műveleteket használt a nagyobb pontosság elérése céljából, ajánlatosnak tartja, hogy matematikai processzorral is rendelkezünk. Ez ugyanis a műveletek többségét 40%-kal gyorsítja. Megszívlelendő, hiszen két 50 K-s állomány keverésekor több mint egymillió utasítást hajt végre a rendszer. Lassú grafikus kártya esetén pedig hosszabb várakozási idővel kell számolnunk.



A SoundFX-Blast Editor grafikus editor SoundBlasterhez. A Silicon Shack cég számos hasonló programot fejlesztett ki: SoundFX, SuperSound, SoundFX-Jr, SoundFX-Voc, SoundCard, SoundBytes, SoundJr és TurboSound.

A SoundFX-Blast segítségével felvehetünk, létrehozhatunk, szerkeszthetünk, módosíthatunk, lejátszhatunk és tárolhatunk hangokat. Semmiféle korlátozás nincs ezzel kapcsolatban: zenét, emberi beszédet, más hasonló természetes hangzásokat vehetünk fel, kezelhetünk. A minőség és az eredeti felvételhez való hűség megtartása már csak rajtunk áll. A rendszer egérrel vagy billentyűzetről vezérelhető, de a fejlesztők egérré optimalizálták — ajánlatos inkább ezt választani.

A rendszer menüvezérelt, párbeszédablakkal. Az főmenüben File, Edit, Play/Record, Modify/Create, Settings, Help menüpontok jelennek meg. Az egyes pontokból aktivizálható műveletek:

File menü — az aktuális hang állomány elemtése állományba akár a jelenlegi néven, akár új néven; már meglévő vagy új állomány megnyitása; kilépés a rendszerből DOS-ba.

Play/Record menü — aktuális hangállomány egyszeri lejátszása elejétől végéig; kijelölt blokk lejátszása; felvett állomány mintavételezési frekvenciájának módosítása; hangerősség beállítása; aktuális teljes hangállomány rögzítése; blokk (hullám kijelölt része) felvétele.

Edit menü — blokk nullázása, törlése; kivágott blokk átvitele/másolása átmeneti pufferbe; átmeneti puffer tartal-

mának beillesztése a hangminta kijelölt helyére; átmeneti puffer tartalmának hozzáfűzése a hangmintához; átmeneti puffer tartalmának keverése az aktuális hangállomány tartalmával, a kijelölt ponttól kezdődően; csatornaváltás.

Modify/Create menü — már létező hangállomány tartalmának módosítása; hullámfrekvencia korlátozása, módosítása szűrővel; visszahangosítás; visszhangok közti időeltolódás beállítása; amplitúdó módosítás (0 ± 40 dB); a hullámforma hangerejének vagy amplitúdójának nemlineáris illesztése; hangállomány megfordítása (mintha fordítva beszélnénk).

A hangok létrehozása a generátor műveletekkel történik. Zajokat, egyszerű dallamokat, összetett hangzásokat, FM-hangokat tudunk így kialakítani. Generálásakor az aktív hangállományt írjuk felül a hullámforma bal oldalán kijelölt ponttól kezdődően a jobb oldali kijelölésig.

Zajgenerátor — a hullámforma kijelölt részét véletlenszerű fehér vagy nagy sávzsílességű zajokkal írja felül. Az amplitúdót mi állítjuk be. A „Filter” művelettel más típusú zajokat is előállíthatunk.

Tiszta szinuszhullámokat, négyszög és háromszög jeleket hoz létre a hanggenerátor, 1 Hz és 20 KHz közötti frekvenciával, beállítható amplitúdóval, az ismétlődési ciklus 1 és 99% között lehet.

Az AM-generátorral amplitúdómodulált szinuszhullámokat állíthatunk elő, megadhatjuk a vivőfrekvenciát, az amplitúdót, a modulációs százalékat. Az FM-generátorral bonyolultabb hangzá-

sokat alakíthatunk ki a frekvenciamodulált szinuszhullámok felhasználásával. Beállíthatjuk a vívőfrekvenciát, az amplitúdót, az eltérési tényezőt. Így a szírehangtól a géppisztolykattogásig a legkülönbözőbb hangeffektusok érhetők el.

A fentiek alapján egyértelmű, hogy amatőr műhelyben akár ez a demó is elég drága műszerek helyettesítésére (szinusz-, háromszög-, négyszöggenerátor). Blokkos üzemmódja alapján Wobbler-generátor imitálására is alkalmas. (Különböző frekvenciájú blokkokat tudunk egymás mögé tenni.)

Adott mintavételezési frekvenciájú hangállományt át tudunk konvertálni más mintavételezési frekvenciájúba.

A SoundFX-Blast editort és a digitalizált hangokat saját szórakozásunkra, oktatásra vagy akár üzleti célra is használhatjuk. Dolgozhatunk SoundFX-Blast hangállományokkal, de bármilyen

más szoftvert által létrehozott hangállománnyal is, feltéve, hogy a 8 bites minták ASCII formátumúak. Szinte korlátlan szoftverlehetőségeket nyitnak meg előttünk a digitális hangok, számításunk azonban arra, hogy időbe fog telni, amíg megtaláljuk a leghatékonyabb technikát az egyedi programokhoz.

Ne feledkezzünk meg arról, hogy a SoundBlaster kártya ugyan tetszőleges hangok kezelésére alkalmas, de természetes hangoknál használható ki legjobban. Azt érezzük természetes hangon, amit kellemes, érthető digitalizált beszédhangként hallunk, nem pedig mesterséges, gépi robothangként.

A lemezen található PLAYSX.EXE program nem része az editornak, DOS szintű indítva önállóan játssza le a SoundFX-Blast hangállományokat. Folyamatossá is tehető ez a lejátszás — tetszőleges billentyű leütésével szakít-

hatjuk meg az ismétlődő dallamot. Több dallam egymás utáni lejátszása is lehetséges. Az egyes billentyűkhöz hozzárendelhetünk billentyűnként 5 hangállományt, 100 ilyen billentyű 500 dallamot reprezentál — ennyit tudunk tehát lejátszani folyamatosan egyetlen utasítással.

A SoundFX-Blast maximális helyigénye DOS 5.0 esetén 350 K. Ez 27 és 82 másodperc közötti felvételnek felel meg.

Konfigurációs igény: IBM PC, 512K memória, matematikai koprocesszor ajánlott, CGA, EGA, MCG, VGA vagy HGA, MS-DOS 2.10 vagy magasabb verzió, SoundBlaster kártya, mikrofon, kábel. A SoundFX-Blast program SoundBlaster kártya nélkül is működik, csak nincs igazán érdekelme — nem tudunk sem felvenni, sem pedig lejátszani.

Verebély Pálné

Writer's Heaven

Gépirók a mennyben

A SolarSoft #277 lemeze, a Writer's Heaven

„...egy olyan billentyűzetdefiniáló program, amely a népszerű PC-Write szövegszerkesztőbe illesztve, azt napjaink leggyorsabb és legkönnyebben kezelhető szövegszerkesztőjévé teszi” — írja Mark Shepard, a program szerzője.

Bár a legtöbb szövegszerkesztőt a tikárságok részére készítik, ritkán gondolnak az ott dolgozó gépirókra. A gépiró munkája közben ugyanis többször kénytelen a gépelt szövegről a funkcióbillentyűkre tekinteni, hogy az egyes parancsokat kiadhassa. Ez a „kapkodás” lelassítja, elkedvetlenülíti, és idegessé teszi. Íme a megoldás: a Writer's Heaven, melyet egy gépiró írt, gépiróknak!

A PC-Write külföldön és itthon is nagyon kedvelt shareware szövegszerkesztő, parancskiadásai azonban nem a gépirók kényelmét szolgálják, ezért gondolta úgy a Writer's Heaven készítője, hogy a PC-Write makrózását lehetővé tartsa használva egy frögépbillentyűzet-központú parancsstrukturát alakít ki. Nincs egy olyan parancs sem a

programban, amelynek kiadásához a gépiróknak el kellene mozdítania a kezét a gépirási pozícióból. Az eredmény: egy gyorsan, hatékonyan, egyszerűen kezelhető és könnyen megtanulható szövegszerkesztő — egy olyan editor, amely lényegesen lecsökkenti a szerkesztés idejét. (Ellenben csak tanult gépiróknak ajánlható.)

A Writer's Heaven is shareware. Ez azt jelenti, hogy mindenki ingyenesen lemásolhatja vagy szabadon forgalmazhatja másoknak — de csak változatlan állapotban, és az összes hozzá tartozó állománnyal együtt. A PC-Write 2.6-os vagy annál későbbi verzióival működött együtt. (A PC-Write 3.03 is megtalálható a SolarSoft Programkönyvtárban.) IBM PC vagy azzal kompatibilis gépeken fut. A billentyűzet „A” betűjével

balra található CTRL, ALT és SHIFT gombokat használja, amelyek az IBM gépeken alapképzésűek (más kiépítésű billentyűzeteken nem ajánlott).

A makróprogram tulajdonképpen egy .DEF(iniál)ó fájl, amely a szerkesztés megkezdése előtt olvassódik be, a billentyűzet újrakiosztását és más utasításokat tartalmaz. A PC-Write kontrollfájljaihoz hasonlóan átférható. Hozzáadja magát a PC-Write parancskészletéhez, így a legtöbb PC-Write funkcióbillentyűt egy másikall helyettesíti az frögép-billentyűzetten. Teljesen eltávolítja a PC-Write WordStar parancsait, amelyek ott voltak definiálva. A font- és speciális karakter-parancsok — az ALT+betű kombinációk — más speciális billentyűkre vannak áthelyezve. Az ENTER/RETURN billentyű SHIFT-STATUS funkciója is módosult. A program monokrom vagy egyszínű CTR monitorral is használható: a betűk sötétek, a háttér pedig világos. A fordított videomód kevésbé veszi igénybe a szemet, mint a normális kijelzés. (Ez a rész minden további nélkül eltávolítható a programból.)

A használat megkönnyítésére a Writer's Heaven tartalmaz egy help képernyőt, amely beilleszthető a PC-Write helpjébe: az első helpképernyőhöz (Basic editing) hozzáféródik a HVN.HLP fájlban található szöveg. Figyelem! Jelenleg a PC-Write automata törléses opciója nem mindig fér össze a Writer's Heavennel. Lassúbb vagy rosszul értelmezett parancsokat, illetve működésképtelen makrókat

eredményez; annál több hibát, minél hosszabb a bevitt szöveg.

De akik lemondanak az automatikus tördelésről, nagyobb sebességgel tudnak majd dolgozni, mert a Writer's Heaven tartalmaz egy tördelési paran-

cstot a sokkal könnyebben elérhető író-gépbillentyűk egyikén.

A PC-Write, mint a legtöbb program, azért nem tökéletes, és akadnak benne felderítetlen részek. S valószínű, hogy a Writer's Heavennél is lehetnek olyan

problémák, amelyekre nem ad választ a lemezen található részletes leírás. (A klaviatúra változását a lemez mellékletünkön elhelyezett klavia.txt fájl illusztrálja.)

Lampert Csilla

„Komplexitás”

A CPLX program a Turbo Pascal egyik hiányosságát próbálja meg pótolni: alkalmas két tetszős szerinti komplex szám összegének, különbségének, szorzatának, hányadosának képzésére, valamint egy komplex szám n -edik hatványának, s 1—20-adik gyökének számítására. Ezekre a lehetőségekre a műszaki élet több területén is szükség lehet.

A komplex számokat kétféle alakban adhatjuk be, illetve láthatjuk a képernyőn. A felső sorban a három szám (Z1,Z2,Z3) a Descartes-féle koordináta-rendszerben látható, míg alul az Euler-formula szerint. Az ablakokat a <SPACE> billentyűvel, illetve az egérrel válthatjuk. A harmadik ablak (Z3) az eredmények részére van fenntartva, ezért azt nem editálhatjuk, ezenkívül a hatványozás s a gyökvonás műveletét sem alkalmazhatjuk rajta (ebben az esetben a program automatikusan az

első ablakra vált). A Z1, Z2 számok értékét EDIT üzemmódban adhatjuk be. Először válasszuk ki a megfelelő ablakot, ezután lépünk be az EDIT módba! Látni fogjuk, hogy ebben az üzemmódban nemcsak változtatni tudjuk a számok értékét, hanem ki is tudjuk cserélni őket egymással (CHG). Miután újra kiválasztottuk az EDIT pontot (E), beadhatjuk a számok kívánt értékét. Használhatjuk a számbillentyűket, valamint az E billentyűt, s a műnuszjelet. Miután beadtuk a szám egyik komponensét, az

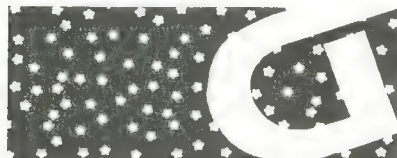
ENTER leütése után beadhatjuk a másik komponensét is. Amennyiben a szám megtölti a rendelkezésre álló helyet, a program automatikusan a következő komponensre ugrik.

A műveleteket a megfelelő műveleti jel beadásával aktivizálhatjuk. A műveletet csak az = jel, illetve az ENTER beadása után hajtja végre a gép, ekkor az eredmény megjelenik a harmadik ablakban (Z3). Hatványozás és gyökvonás esetén a program várja a kitevő értékét (x=). Gyökvonáskor x maximum 20 lehet! A műveletet mindig az aktuális számon végzi el a program (kivéve a 3. ablak). Hatványozáskor az eredményt Z3 mutatja, míg gyökvonáskor egy külön képernyőn láthatjuk a gyökök értékeit egymás alatt.

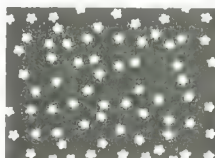
A program bármelyik üzemmódjából az ESC billentyűvel, illetve az egér jobb gombjával tudunk alapállapotba visszatérni. Az üzemmódokat, műveleteket, ablakokat az egérrel is ki tudjuk választani, s a bal gombbal aktivizálni tudjuk. A programból a QUIT módban tudunk kilépni.

A grafika VGA/EGA monitorra készült.

Záruba Károly



GALAX
KERESKEDELMI KFT.
1113 BUDAPEST, BÖCSKAI UT 54
TELEFON FAX 161 08 57



3M, írásvetítőre helyezhető LCD kivetítő panelek 5-féle típusban, 107 cFt-től, amellyel számítógépének monitorképét akár 4x4 m-es méretben is láthatja. 3M írásvetítők 9-féle típusban, hordozható és alulrólvilágító kivitelben.

Bemutató ● Szaktanácsadás ● Kipróbálási lehetőség ● Bérleti

Minőséggel és szakértelemmel várjuk Önt!



3M 2770T
Overhead Projector

IBM PC

SOLARSOFT
KATALÓGUS

A programok ára:
lemezenként 399,- Ft + áfa

Értékesítés:
FLOPPYLAND
Budapest V., Váci u. 84.
Telefon/Fax: 118-2651

Cédus Karolina Áruház
Budapest XI., Karolina út 17.
Tel.: 166-2111 • Fax: 185-2221

Lemezszám: 500

Név: C++ Toolkits #3

Szerző: Többek, 1989, 1990.

Leírás: OOP-kiegészítések
A lemez 20 tömörített állományt tartalmazunk:
BAM — (BAM — Bidirectional Associative Memory) Zortech C++ 2.0 nyelvén, OOP technikával implementálja a BAM-konceptiót, amelyet Bart Kosko fejlesztett ki. Patrick Simpson bővítéseit is figyelembe vették. BGICLA — (BGI — Borland's graphics Interface) Turbo C++ grafikus rajzoló programok, forrással. Tanulságos!
BROWSE és BROWSE2 — C++ böngészők forráskóddal.
BRENT — Hash algoritmus C++-ban, forráskóddal. R. P. Brent rendezési ötletének megvalósítása mind pszeudokódban, mind C++ nyelvén. Brent technikája azon a felismerésen alapul, hogy egyes alkalmazásoknál (ilyenek például a fordítók) a sikeres keresések száma messze meghaladja a beillesztések (insert) számát. Az ő módszere több időt szán a beillesztésekre, mert így esetleg kedvezőbb helyzetek állnak elő.
CCLASS — Osztályöröklés C++-ban, forrással.
COLLEC — Gyűjtő osztály és mintaprogram, amely összeszámolja a kulcsszavakat és az azonosítók egy forrásfájlban. Jó példa a többszörös öröklésre, bemutatja a tagokra mutató pointerok használatát.
LITERA — AWK alapú preprozessor, amely nagyban lecsökkentheti a C++ kód dokumentálási nehézségeit. Szöveg és források.
LLIST — „Linked List” osztály C++-ban. Megmutatja, hogyan kezdünk építeni egy tároló osztályt C++-ban. (Három forráslista)
MENU — Menürendszer teljes C++ forrással (6 program).
MOD2CP — A Modula-2 és a C++ összevetése, különös tekintettel a komplex típusokra, 5 forrásprogrammal és a tesztprogram forráslistájával.
NDEL — Az utasítássorban megadott fájlokban kívül mindent töröl. A törölendő állományoknak ugyanabban a könyvtárban kell lenniük, ahol az NDEL program található. „*” használata tiltott. (Forráskód és maga a futtatható program.)
PARSE — LARL(1) nyelvtan C++-hoz. Figyelemre méltó a konstruktor/függvény definíciós probléma megoldása. Több forrásprogram.

PARSER — Gyakorlatilag ugyanazokat a forrásokat tartalmazza, mint a PARSE, csak egyetlen állományban.
RTRACE — Sugárkövetés C++ nyelvén, forráskóddal.
STARTUP — A Turbo C++-hoz rendszer-startup kód. Figyelem! Kicsomagoláskor feltétlenül adjuk meg a „d” opciót, máskülönben egymás hegyére-hátára ír mindent!
TASK — Multitasking C++-ban. 6 forrásprogram — taszkok közötti üzenetátadás, szemafor osztály kezelése, lock megszüntetése ...
A tesztprogram forráskódban és végrehajtható formában is megtalálható.
TAWK — AWK interpreter erősen kommentezett C++ forrással, főleg a különböző típusú ASCII állományok kezelésére, nyomtatására, 11 forrásprogrammal.
TXTSCR — Egyszerű kiértékelő rendszer C++-ban, komplett forrással.
ZELLER — Zeller 1987-os módszerét kódolták C++-ba. Annak megállapítása, hogy egy naptári hét hányadik napja egy adott nap, igen egyszerű lenne a C++ mktime() függvény felhasználásával. Így ennek a forráskódnak inkább elméleti a jelentősége.

Lemezszám: 501

Név: C++ Toolkits #4

Szerző: Többek, 1989, 1990.

Leírás: OOP-kiegészítések
Példa arra, hogyan alkalmazzuk a BISON és a FLEX csomagot saját nyelv készítésére C++-t, illetve C kódot használva.
A BISON és a FLEX csomagok külön lemezeken vannak, nehogy az állományok összekeveredjenek. Ez az #501 lemez a BISON-t tartalmazza, a FLEX pedig a SolarSoft #502 lemezen található.
A COM.ZIP tartalmával helyettesíthetjük a hiányzó FLEX/BISON állományokat.

Lemezszám: 502

Név: C++ Toolkits #5

Szerző: Többek, 1989, 1990.

Leírás: OOP-kiegészítések
Az #501 lemezzel együtt használható!

A lemezen 6 tömörített és két szöveges állomány található.

BROWSE — Az aktuális könyvtár *.c, *.cpp és *.hpp állományait végignévez felsorolja a használatban lévő objektumokat.

DATDECL — Egyszerű C++ osztály dátumtípusokhoz. Összeadáshoz, kivonáshoz, dátumok beolvasásához és nyomtatáshoz szükséges függvényeket tartalmaz.

FLEX — LEX bővítés. Az #501 lemezen található COM.ZIP és BISON.ZIP állományokkal együtt használjuk. Saját fordítót írhatunk vele. A nyelvi kifejezések nyelvtani szabályait és a rutinokat átalakítja C nyelvű függvényekké és táblázatokká. Stringeket olvas be, nyelvi elemekké alakítja ezeket.

OVLOAD — Függvények betöltésével kapcsolatos tesztkészlet. (14 forrásprogram). A függvények adatkonverziót, hibakezelést végeznek.

PARSER — C++ böngésző forrásal.

PCTECH — A #340 lemezen található HyperHelper hipertexthelp-olvasó egy friss verziója (2.01a) található ebben a tömörített állományban. Ezenkívül a PC TECHNIQUES magazin tartalmából kedvezményesen: 10 C (közte 2 C++) forrásprogram, 6 Pascal 2 asszemblér (ellenőrző összeget készítenek állományokhoz) és egy Basic forrásprogram.

A két szöveges állomány:
CPULUSS.ASC — Scott Robert Ladd String osztályai C++-ban. Különböző megközelítésekben tárgyalja a C++ sztringeket.
OORULE — 50 hasznos tipp a C-programozóknak, ezek alapján könnyebb lesz az OOP-technika elsajátítása a C++ nyelvet használva.

Lemezszám: 503

Név: Turbo C Toolkit

Szerző: Többek

Leírás: Hasznos segédprogramok — 10 tömörített állomány, két tömörített C nyelvű forrás, egy .EXE állomány.

CSOURC 1.5 — C nyelvű forrásprogramok szervezése, kereszterferencia készítésével. A kiválasztott programok változóit, függvényeit, azonosítóit sorolja fel, kivághatunk programrészeket, az outputok képernyőre, nyomtatóra, lemezre állományba irányíthatók. Menüvezérelt, jól dokumentált, de

a forrást nem mellékelte; szerzője Randy Minder, USA.

DBF — Turbo C 1.0 rutinok dBase III kapcsolathoz. 12 forrásnyelvű rutin, amelyek állományokat nyitnak, zárnak, új állományokat hoznak létre, rekordokat kezelnek.

Nem foglalkozik indexállományokkal, adatpuffereléssel, memomemókkal. Jól dokumentált, a szükséges könyvtárt is tartalmazza. Szerzője Mark Sadler, USA.

DIRESEL — Függvény, forráskóddal. Megadott állománynév keresése után egy mutatót ad vissza.

Szerzője Vernon E. Davis, USA.

DISKLI — Lemezkatálogosizáló program forráskóddal. Mivel lemez vagy floppy könyvtárakat, állományokat katalogizálja, visszakérdezhajuk állománysorrendben vagy lemezegységként. Forráskódot is találunk, de ez csak tanulmányozásra alkalmas, a szerző

nem mellékelte saját ablakait, sem a képernyőkezelő rutinok forrását. Csak CGA és EGA grafikus kártyával működik. Szerzője Marvin J. Cook, USA.

ECSRC — Nagyon gyors videourutinok, forrással. A 0-7 videomódot támogatják, minden képernyőre írt output után aktualizálják a kurzorpozíciót. CGA-ra készültek. Szerzőjük Joe Odenwelle, USA.

EXE2CO — Az EXE2BIN megfelelője. Szerzője, Chris Dunford akkor fejlesztette ki, amikor a DOS 3.3 verziótól kezdve az EXE2BIN már nem került az alapsomagba, hanem külön meg kellett volna vásárolni a Technical Reference részeként. Teljes dokumentáció, forrásszöveggel.

IBMCOM — IBM RS232 (interrupt vezérlésű, soros kommunikációs) rutinok COM1 és COM2 portra. A shareware verzió a small modell könyvtárát tartalmazza. Az input puffer 2000 karakter/port kapacitású lehet, a puts_com() rutin output puffere pedig 80 karakteres. A sebességet 300 és 112 000 baud között állíthatjuk. Jól dokumentált, forráskódot nem tartalmaz. Fejlesztője az Intrinsic Computer, USA.

INDEX — ISAM fájlkezelő, Turbo C 1.0-ban. Lehetővé teszi, hogy indexállományokat kezeljünk C nyelvű programokból. Egykulcsú állományokat kezel, a kulcs típusa azonban tetszőleges lehet: lebegőpontos, dupla pontosságú, vagy akár a felhasználó által megadott adattípus. Forráskóddal (10 program), könyvtárral, dokumentációval. Szerzője Jim Mischel, USA.

SCRLIB — Turbo C-ben gyors képernyővezérlést tesz lehetővé. Az ANSI.SYS-t gyors kurzorvezérlő

rutinokkal tette hatékonyabbá a szerző, Paul Raulerson. Ezeket egy könyvtárba gyűjtötte össze. Használatához az ANSI.SYS-t installálnunk kell a CONFIG.SYS állományban.

TCDEBU — Debugger Turbo C nyelvű forrásprogramokhoz. Jól dokumentált, de sajnos a forrást nem tartalmazza. Szerzője L. David Baldwin.

A két tömörített program:
RMTREE — Teljes alkönyvtár eltávolítása. Rejtett és rendszerállományokra rákérdez.

CCC — Nyomatóprogram, amely a C kódot blokkosítva nyomtatja ki a jobb olvashatóság kedvéért.

(Forráskód nélkül.)

NUMOFF — Rövid forráslista a NumLock gomb kikapcsolására.

Lemezszám: 504

Név: PC-Type II 2.0

Szerző: Richard W. Earl, Buttonware Inc., USA

Leírás: Szuper szövegszerkesztő A PC-TYPE II 2.0 nagy kapacitású, nagygyorsú szövegszerkesztő három lemezen. Kibontva 2 Mb-ot foglal el a merevlemezben.

A 29 fejezetből álló dokumentáció 741 kb-ot foglal.

Főbb szolgáltatásai a teljesség igénye nélkül:

— Kőrlevelet készíti. Különböző állományok részeit vághatja össze az INCLUDE és IF utasítások segítségével.

— Szótár 100 000 angol szóval, a bejegyzések módosíthatók, lacserelhetők, új szavakat vihetünk be. A helyesírás ellenőrzésekor a hibás szavak helyett helyes hasonlót ajánl fel, észreveszi a véletlen szóismétléseket. Keresztjeletvejtést segítő programot is tartalmaz.

— Számos, különböző típusú grafikont készíthetünk a táblázatba foglalt adatokból.

— Hexa számokkal is végez műveleteket.

— Vonalak és kereteket is rajzol. — Sorszámozás és stringtöbbszörösítés, beillesztés adott sorba.

— Visszahozhatjuk a törölt vagy rosszúl módosított sorokat.

— Adatbázisok és táblázatkezelők adatai bevehetők, főleg a cég saját fejlesztésű szoftvereiből: PC-File, PC-File:DB és PC-Calcul. (Soros állományú ASCII adatbázisok és vesszővel elválasztott álló-



Jelenleg mintegy 1000 szoftverből, illetve
külföldi szakkönyvből válogathat.

Az aktualizált lista megtalálható
mostani számunk lemezmelletlenén.

A megrendelt szoftvert vagy külföldi szakkönyvet
postai utánvétellel 2 héten belül házhoz szállítjuk.

MEGRENDELÉS

Megrendelem postai utánvétellel az alábbi termékeket.
A vételárát és a postaköltséget átvételekor kifizetem.

A) SZOFTVEREK:

.....
.....
.....

B) SZAKKÖNYVEK:

.....
.....
.....

Dátum:

(aláírás)



Amíg a készlet tart...

Megrendelem az Alaplap kiadványsorozataiban (Könyvek,
Füzetek, Lemezek) eddig megjelent (és még kapható) mű-
vek közül az alábbiakat:

ALAPLAP KÖNYVEK

... pld: Nagy Gábor: Tömör gyönyör, 256,- Ft

... pld: Kis János-Szegedi Imre: Új vírusvédelem,
256,- Ft

... pld: Kis János-Szegedi Imre: Vírushatározó, 256,- Ft

ALAPLAP FÜZETEK

... pld: Detrik Péter: Az SQL nyelvről, 375,- Ft

ALAPLAP LEMEZEK

... pld: Bliss főkönyvi program demója és leírása,
750,- Ft

... pld: Norton Guide keretprogram leírása, 500,- Ft

... pld: PathMinder segédprogram leírása, 1000,- Ft

... pld: CSProlog nyelv leírása, 1000,- Ft

... pld: LIM EMS 4.0 memóriakezelő leírása, 1000,- Ft

... pld: Windows 3.0-hoz magyar betűkészletek, 1000,- Ft



PC Turbo Klub VÁLASZLAP

Eddigi PC Turbo Klub-tagoknak nem kell kitölteniük
és beküldeniük, ha címüket az új címadatbázisban is
szerepeltetni akarják. (Lásd erről részletesebben cik-
künket az 54. oldalon.)

☐ A PC Turbo Klub tagja vagyok, de kérem, hogy
az új címadatbázisban címem már ne szerepel-
jen.

☐ Nem vagyok a PC Turbo Klub tagja, de szeret-
ném, ha túloldalt megadott címemet az új cím-
adatbázisba felvennék.

Tudomásul veszem, hogy a PC Turbo Klub új cím-
adatbázisába való felvétel esetén címemet az Alap-
laptól szakmai címanyagot kérő cégek és intézmé-
nyek direct mail akcióikhoz (meghívók, prospektu-
sok stb. kiküldéséhez) felhasználhatják.

Dátum:

.....
aláírás



INFORMÁCIÓKÉRÉS

Kérem, hogy
az itt általam
**BEKARIKÁZOTT
KÓDSZÁMÚ**
hirdetésekkel
kapcsolatban
küldjenek
részemre
bővebb
tájékoztatást.

Beküldhető:
1992.
szeptember
30-ig

01	02	03	04	05
06	07	08	09	10
11	12	13	14	15
16	17	18	19	20
21	22	23	24	25
26	27	28	29	30
31	32	33	34	35
36	37	38	39	40
41	42	43	44	45
46	47	48	49	50
51	52	53	54	55
56	57	58	59	60
61	62	63	64	65
66	67	68	69	70
71	72	73	74	75
76	77	78	79	80

ALAPLAP
1992/8
AUGUSZTUS

FELADÓ:**A) Egyéni érdeklődő:**

Név:

Utca, házszám:

Helység:

Irányítószám:

B) Vállalati érdeklődő:

Cégnev:

Ügynökség:

Utca, házszám:

Helység:

Irányítószám:

Telefon/Fax:

**FELADÓ:**

Név:

Foglalkozás/Beosztás:

Cím:

Helység:

Irányítószám:

**Cédrus Kiadó**
Pf. 74**Budapest****1441****Cédrus Kiadó**
Pf. 74**Budapest****1441****FELADÓ:**

Név:

Cég:

Utca, házszám:

Helység:

Irányítószám:

Telefon/Fax:

FELADÓ:

Név:

Cég:

Utca, házszám:

Helység:

Irányítószám:

Telefon/Fax:

**Cédrus Kiadó**
Pf. 74**Budapest****1441****Cédrus Kiadó**
Pf. 74**Budapest****1441**

A LEMEZMELLÉKLET TARTALMA:

- ☐ „Biliárdozó marslakók” — avagy a lokális modulok
- ☐ Multitask randevú-mechanizmus
- ☐ Órajelmegszakítás-számláló
- ☐ Szóhosszúsági statisztika-készítés Snobollal
- ☐ Alapműveletek komplex számokra
- ☐ A kínai számneves feladat megoldása
- ☐ A Gerenia kézírásfelismerő rendszer demója
- ☐ Tili-toli Herculesen (és a forrásprogram)
- ☐ Alaplap Posta — szoftverek, szakkönyvek kínálata
- ☐ Új Windows alkalmazások és játékok a SolarSoftban



Az innováció Önnek dolgozik™ **3M**



Ne csak floppyt a FLOPPYLAND-ból!

Tisztelettel értesítjük kedves vásárlóinkat, hogy beindítottuk minőségil hardverek forgalmazását.

- Holland Tulip számítógépek professzionális felhasználóknak
- 3M írásvetők már 38.000 Ft-tól
- 3M színes LCD kijelzők 110.000 Ft-tól
- Transpack hordozható és cserélhető SCSI hard diszkek **120-535 MB-ig**
- Sound Galaxy NX hangkártya csak 19.900 Ft

11 csatornás FM szerelő szintező 4 wattos erősítővel, hangszórókkal.Sound Blaster, AdLib, Covox, Speech Thing, Disney Sound kompatibilis. MIDI, CD ROM interface, Game port, DAC és ADC modulok lejártszáma, és digitalizáláshoz. Szoftveres hangrő, hangszin, balansz szabályzók.



Corel Draw 3.0 27.900 Ft, upgrade 15.900 Ft

Árunk ÁFA nélkül értendő!

Cédrus Floppyland Kft Bp. 1056 Váci utca 84. Tel./Fax: 118-2651



K&Szo Kft.

Szabkolt: 1055 Bp. V. Néphadsereg u. 6.

Tel./Fax: 111-8268

Tel.: 132-8717

Nyári, forró slágereink:

Fontmenger for Windows (TrueType fonteditor)	18.000
Publisher's PowerPal (MS Word 5.x, Works 2.0 és WordPerfect 5.x magyartása bármely mátrix/lézer nyomtatóra, skálázható fontokkal)	8.400

386MAX 6.0 (MS C/C++ 7.1HEZ)	9.900
ACEFILE FW (BASE COMP.)	30.200
ADOBE ILLUSTRATOR 4.0	69.900
AFTER DARK 2.0 FW	5.000
ALLTYPE (TRUE TYPE KÖNYV)	8.400
AUTOMAP ROADATLAS EUROPE	14.000
BITSTREAM FACELIFT 2.0	9.900
BITSTREAM MAKEUP FW	15.000
BLINKER 2.0	33.600
CODE BASE 4.5	39.900
COREL DRAW 3.0	26.300
DESQVIEW 386 V2.4	21.000
DESQVIEW/X 386	25.000
DR DOS 6.0	11.500
EXPERT HELP/HYPertext (NO)	15.800
FLOWCHARTING 3 V2.0	30.500
FOXPRO 2.0	68.000
FOXPRO 2.0 LAN (6 USER)	79.000
FRAMEWORK IV	54.000
HARVARD GRAPHICS 3.0 UPRGR	19.000
IBM OS/2 2.0	19.900

IBM OS/2 2.0 UPRGRADE	15.000
LAN ASSIST PLUS 3.1	37.800
LAPLINK PRO 4.0	15.800
MATHCAD 3.1 FW	47.300
MS C/C++ 7.0 & SDK 3.1	49.000
MS C/C++ 7.0 & SDK 3.1 UPRGR	29.900
MS COBOL 4.5	85.000
MS EXCEL 4.0 FW	46.000
MS EXCEL 4.0 FW COMP. UPRGR	18.000
MS FORTRAN 5.1 FW	42.000
MS MOUSE SERIAL	11.500
MS MULTIMEDIA BOOKSHELF	19.000
MS PROJECT FW 3.0 UPRGR	22.000
MS TEST FOR WINDOWS	39.900
MS VISUAL BASIC FW	13.000
MS WINDOWS 3.1	14.000
MS WINDOWS 3.1 UPRGRADE	8.000
MS WORD 5.5+GRAMMATIC	37.800
MS WORD FW 2.0	45.000
MS WORD FW 2.0 UPRGR	18.900
MS WORKS FW	18.900

MS WORKS FW UPRGR	11.000
METAWARE PASCAL DOS/UNIX	99.000
NORTON DESKTOP FW 2.0	15.800
NORTON DESKTOP FOR DOS	17.900
ORBITS	5.300
PC-SIG LIBRARY (CD-ROM)	29.000
PC TOOLS 7.1	16.000
QEMM 386 v6.0	9.900
QUATTRO PRO 4.0 COMP. UP	14.000
SOUND BLASTER 2.0	15.000
SOUND BLASTER PRO 2.0	27.000
TOPSPEED MODULE C, C++, PAS	27.300
WATCOM C V9.0/386	87.000
WORDPERFECT FW UPRGR	16.000
WORDSTAR 7.0 UPRGRADE	13.000
WORDSTAR FW UPRGR	20.000
PINNACLE MICRO CD READER	35.000
VIDEO PLUS (VIDEOZÁS PC-N)	70.000
ZOONFAXMODEM 9624	19.000

FW = for WINDOWS

Szenzáció!!!

MAGYAR KÖNYVEK - INGYEN!

A KÖVETKEZŐ SZOFTVEREK VÁSÁRLÓI INGYENESEN MEGKAPJÁK AZ ADOTT PROGRAMRÓL BOTTI MAGYAR NYELVŰ ISMERTETŐT, SZAKKÖNYVET.

MS WORD 5.5, CLIPPER 5.01, MS WORD FOR WINDOWS 2.0, FOXPRO 2.0, BORLAND C++, NOVELL, MS WORKS, QUATTRO PRO, COREL DRAW, POWERBASIC

KÉRJE ÁRKATALÓGUS LEVEZŐNKET!

Árunk a 25%-os áfát nem tartalmazza.

ADATMENTÉZÉS

(Meghibásodott winchesterekről)

K&RT KFT
WINCHESTER CENTRUM

TEL.: 181-0539

186-5477

FAX: 161-1211

**ÉRTÉKESÍTÉS
JAVÍTÁS**

1119 Budapest
XI., Fehérvári út 55.
ÁÉB 204-10229



mányok — Wordstar — is megfelelők.)

— 10 különböző állománnyal dolgozhatunk 4 egyidejűleg megnyitott ablakban, ugyanannak az állománynak különböző részeit szerkeszthetjük az egyes ablakokban.

— Az egyes állományokba négy különböző helyet jelölhetünk ki ún. könyvjelzővel, ezek áthelyezhetők. — Makrók: egyszerű 25 makró lehet aktív. A makrók hívhatnak más makrókat, illetve önmagukat. — Tabulátor, lapszél, bekezdés többféle is lehet: egyszerűes, kétszeres, háromszoros.

— Lapszámozás; megadható (lehet feltételes vagy automatikus). — MOVE és COPY utasítások vonatkoznak sorokra, bekezdésekre, mondatokra, szavakra, sőt szövegblokkokra (állományok között is).

— SEARCH és REPLACE utasításoknál megadhatjuk azokat az oszlopokat is, amelyekre a keresés kiterjedjen.

— ALIGN és SORT utasítások adatszlopokra vonatkoznak.

— GET és EDIT utasítások vonatkoznak más szöveges állományok részeire vagy egészére.

— SAVE — szerkesztés közben is elmenthetjük az állományok egyes részeit, vagy a teljes állományt. Több mentési másolatot is készíthetünk.

— HEADERS / FOOTERS — Módosíthatók — kikapcsolhatók. (Idő, dátum, lapszámozás, páros/páratlan oldalak megadása.)

— PRINT — Állományok kijelölt részeit is kinyomtatjuk. 19 ESC szekvenciát definiálhatunk nyomtatónkhoz.

— A PC-LABEL-t tartalmazza, így gyorsabb a címkefeldolgozás. — Környezetérzékeny help rendszer.

— Csaknem minden tevékenység menüvezérelt.

— Az utolsó 10 utasítást tárolhatjuk — későbbi ismételt felhasználásra.

— Általában rendelkezésre állnak a DOS-utasítások a PC-Type II-ből, kilépés nélkül.

— Tetszés szerint konfigurálható, teljesen rugalmas.

Lemezszám: 505

Név: Galaxy Lite 1.6

Szerző: Starlite Software, USA

Leírás: Szövegszerkesztő

Az Omniverse a közkedvelt Galaxy 2.43 verziója után a 3.0-t már nem terjesztette shareware-ként. Azokra a felhasználókra tekintettel, akik megszerették a Galaxyt, és igényelnék a további verziókat is, a Starlite Software kifejlesztette Galaxy Lite elnevezésű szövegszerkesztőt.

Vezérlése történhet menüből vagy a billentyűzetlen begépelés gyors utasításokkal. Az első módszert a kezdőknek ajánlják, míg a második a gyakorlottabbak számára kedvezőbb. A rendszer utasítás-készletének kialakításakor azt vették figyelembe, hogy miként szokták használni az emberek a szövegszerkesztőket. Ezáltal olyan rendszert sikerült létrehozni, amelynek kezelése fél óra (!) alatt elsajátítható. 70 000 szavas szótár, helyesírás-ellenőrzés.

Alt és Ctrl billentyűkombinációkkal aktivizálhatók a formázó, kurzorpozicionáló és a undelete (karakter vagy sor) utasítások.

Funkciós billentyűkhöz rendelték a leggyakoribb parancsokat (18).

Nyomatás írásképe (betűtípusok, aláhúzás, fejszóveg, lábjegyzet, ...) beállítható.

Egyidejűleg szerkeszthetünk és nyomtathatunk a rendszerrel. Makrózás lehetséges — jó makró-készlet, saját makrók írása is könnyű.

Az ablakrendszerrel két állomány közti kivágás/beillesztés valósítható meg.

A zoom lehetővé teszi, hogy bármikor teljes képernyőre nagysítsuk ki a kiválasztott ablakot.

Ismeri a WordStar parancsokat, elfogadja, kezeli a WordStar állományokat.

Lemezszám: 506

Név: ProLite 1.01

Szerző: Quicksoft, USA, 1989.

Leírás: A PC-WRITE könnyített változata

A PROLITE vagy PC-Write Lite a közkedvelt PC-WRITE szövegszerkesztő program kisebb kaliberű változata. Azok számára fejlesztették ki, akiknek laptop gépeiken nincs szükségük a PC-WRITE összes szolgáltatására. Programozók, újságírók, fordítók, diákok például semmit nem fognak hiányol-

ni, ha ezt használják PC-WRITE helyett.

A két program adatformátuma és billentyűparancsai azonosak, így akár felváltva szerkeszthetünk egy szöveget a PC-WRITE és a PRO-LITE segítségével.

A program már 256 kb-át memóriában is fut (a helyesírás-ellenőrző nélkül), szótára bővíthető, saját nyomtatóinkhoz akár egyedi vezérlőket is készíthetünk.

Lemezszám: 507

Név: PC-BROWSE & NG MAKER

Szerző: Bob Wallace, Quicksoft, USA, 1990. (PC-BROWSE) Steven W. Kurtz, USA, 1990. (NG MAKER) John C. Gordon, USA, 1988. (NC_SPLIT)

Leírás: Hipertext és fájlböngésző + NG dekompile

A lemezen két fő program található. A PC-BROWSE egy univerzális szövegkeresővel kombinált, razidensön is használható hipertext rendszer, az NG MAKER csomag pedig a Norton Guide formátumú és a PC-BROWSE formátumú adatbázisok tervezéséhez ad segítséget.

A PC-BROWSE az első olyan elterjedtebbnek számító hipertext rendszer, amellyel nemcsak hogy a képernyőre lehet gyorsan kilátni az adatbázis megfelelő oldalait, hanem a program keresőfunkcióját kiegészítve a megtalált szövegek közül tetszőleges blokkot is ki lehet jelölni, majd azt az éppen futó szövegszerkesztővel szerkesztett szövegbe be lehet másolni, illetve ki is lehet közvetlenül nyomtatni. (A lemezen található egyik mintapélda a Hamlet „Lenni vagy nem lenni” monológja!)

Az NG MAKER program segítségével szöveges forrásállományainkat készíthetjük elő mind a PC-BROWSE, mind pedig a Norton Guide programok számára.

A lemezen található még az NG_SPLIT program, angol nyelvű leírással és C nyelvű forrással. Ez az egyik legtekélyesebb Norton Guide-visszafejtő program, nemcsak a forrásszövegeket fejtí vissza teljesen komplett és szerkeszthető formába, hanem előállítja a Guide újáépítéséhez szükséges batch és a linker paraméterezését biztosító listafájlt is.

Lemezszám: 508

Név: Proque Lite 1.0

Szerző: FormalSoft, USA, 1990.

Leírás: A QUBECALC 3D továbbfejlesztett változata

A közkedvelt háromdimenziós táblázatkezelő, a Qubecalc alaposan továbbfejlesztett változata található ezen a lemezen. A program jelen verziója gond nélkül írja-olvasza a Lotus, DBASE és a megfelelő szerkezetű ASCII állományokat. A maximális kockaméret is alaposan megnőtt: most már 512*512*512 cellából álló munkaterülettel gazdálkodhat a program felhasználója, ami messze több az irodai feladatokhoz szükségesnél. A lemezen — természetesen alaposan összetömörítve — több mint 400 kb-át méretű dokumentáció található a program parancsait, menüit, makróit és használatát. (Ez 186 oldalas kézikönyvet jelent.)

A Lotus 1-2-3 szinte minden függvénye itt is megtalálható, ráadásul a munkakocka mind a hat oldaláról szemügyre vehetjük és kezelhetjük három dimenzióban elhelyezkedő adatainkat. Állományok összekapcsolására is van lehetőségünk. Rendezettségét 9 kulcsig kezel.

Makrónyelve könnyen elsajátítható. Egérrel is vezérelhető a rendszer.

Lemezszám: 509

Név: ZIPVIEW/LHA2.13/LHAUTO

Szerző: FAO Productions, Hollandia (ZIPVIEW) Haruyasu Yoshizaki, Japán (LHA) John A. Qualtrough, Q-Ware, USA, 1991. (LHAUTO)

Leírás: Tömörítő

ZIPVIEW — Kellemes, a Norton Commanderhez készített, de önállóan is használható kukkantó és keretprogram kibontáshoz. .ZIP, .LZH, .PAK, .ARC, .ZOO és .DWC kiterjesztésű állományokat kezel. Sajnos a dokumentáció holland, de a program angol nyelven üzen, így teljesen érthető. LHA 2.13 Eredeti csomagolásban (ellenőrizhető — az LHA LHA213.EXE parancs adjuk Yoshi mesterművének legfrissebb változatát).

LHAUTO — segédprogram (a felhasználó számára) egyetlen lépésben készíti el az SFX archívot a forrásállományokból az LHA segítségével.

A lemezen található NCMAIN.EXE állományra kell lecsatolni a Norton Commander központi állományát.

Lemezszám: 510

Név: ARJ 2.30

Szerző: Robert Jung, 1992.

Leírás: Tömörítő

A német Computer Persönlich című lap tesztjében ez a nálunk is méltán népszerű program legfrissebb verziója, a 2.30-as került az első helyre. Olyan vetélytársakat előzött meg, mint például a PKZIP és az LHA.

A tömörítők tesztelésekor a tömörítési hatásfokát, sebességét, a dokumentáltságot és a szolgáltatások sokrétűségét vették figyelembe. Az ARJ hatékonysága legjobban az adatbázisok, tömörítetlen grafikus állományok és a nagy terjedelmű dokumentációk tömörítésekor érvényesült. Volt rá példa, hogy 9 Mbájtos adatbázist 800 kb-ára tömörített!

Az ARJ az egyetlen olyan zömítő, amely többkötetes, együttes archiválásra is képes. Ez azt jelenti, hogy floppyra közvetlenül archivál — függetlenül attól, hogy hány darab és mekkora inputállományról kell dolgoznia. Még további lemezterületet sem igényel. Ez a tulajdonság teszi alkalmassá az ARJ-t (DEARJ-t) arra, hogy nagyobb szoftvercsomagok terjesztésének ideális eszköze legyen.

Nem kell ugyanis arra ügyelnünk, hogy egy-egy floppyra teljes állományok férjenek fel. Amikor szükséges, az ARJ automatikusan feladja az állományokat, kicsomagoláskor pedig összerakja a megfelelő részeket. Ezen szolgáltatása révén válhat az ARJ a „szegény ember” backup-segédprogramjává. Az ARJ-vel készített archív állományok tartalmazzák az elérési útvonalakat, az állományok dátum- és időadatait és a fájlattribútumokat. Indexállományt is létrehozhatnak, amely az egyes kötetekre vonatkozó információt tartalmazza, visszakapja.

Magas szintű megbízhatóságot nyújt az ARJ-archívok adatszerekezete: a fejléc (header) felépítése és a 32 bites CRC. Az archiválási

folyamat közben is ellenőrizhetjük a tömörítést. Más hasonló programok kitérnek az inputállományt, mielőtt az archív állományt helyességéről meggyőződhetnénk.

A „security envelop” lehetőséggel leltárlhatjuk (lock) az archív állományt, így ezt az ARJ már nem tudja többé módosítani.

Lemezszám: 511

Név: 4EDIT & 4ZIP

Szerző: Többek

Leírás: Kiegészítések a 4DOS programhoz

A SolarSoft #383 lemezen levő 4DOS programhoz gyűjtöttünk össze néhány igen hasznosnak látszó kiegészítést erre a lemeze, négy tömörített, önkesomagoló állományba:

4DOSMIC — Egér meghajtók Microsoft és Logitech egértípusokhoz 4DOS-ban. Szerző — Tom Ordelman, European Cybernetics, Hollandia.

4DOSPAK — Megjegyzéseket, leírásokat fűzhetünk az állományokhoz ezzel a segédprogrammal. Szerzője Frans Veldman, Hollandia, 1990.

4UTILS — Két segédprogramot tartalmaz. COPYD szerzője David P. Lemire, 1989. Állományok másolására alkalmazható. A KBFX2 3.0 verzió rezidens, billentyűzettel kapcsolatos bővítmések tesztelhető. Szerzője Skip Gilbrech, 1986.

4ZIP ED — négy segédprogram: — 4EDIT 1.03 — Teljes képernyős editor a 4DOS fájlleírások szerkesztéséhez. Szerzője Iram J. Weinstein, USA, 1990.

— 4ZIP 1.04 és 4UNZIP

A 4ZIP arra szolgál, hogy a 4DOS-nál megszokott kommentjeinket ne veszítsük el, ha a PKZIP/PKUNZIP-pel (SolarSoft #421 lemez) dolgozva tömörítjük ritkábban használt állományainkat. Maga a 4ZIP program nem tömörít, csupán kiegészíti a PKZIP és PKUNZIP már megszokott szolgáltatásait. Szerzője Iram J. Weinstein, USA, 1990.

— XDIF 3.0 — Megjegyzésekkel ellátott könyvtárakat kezelő segédprogram. Szerzője Iram J. Weinstein, USA, 1990.

A 4DOSCOLS szöveges állomány elmagyarázza, hogyan oldhatjuk meg 4DOS-nál a színcserét.

Tárgyszerű víziók

A Borland International Object Vision 2.0 for Windows (továbbiakban OV) programja nem más, mint egy Windows-alapú alkalmazásgenerátor — elsődlegesen adatbázis-kezelő alkalmazások „díszlettervezéséhez és megjelenítéséhez”.

Az OV nem igényel programozói kvalitásokat, nem kell hozzá programfejlesztői véna; bárki látványos és korrekt (azaz hibátlan) adatbeviteli, adatkarbantartó és lekérdező képernyőket csinálhat vele dBASE- vagy Paradox-kompatibilis adatbázisokhoz.

Az OV ikonszervezést, barátságos kezelői felületét kihasználva teletömte a programot sok kényelmi funkcióval. Bár billentyűzetről is rövid úton elérhető bármilyen funkció, egérrel a komplexebb szolgáltatások nagyobb része azonnal aktivizálható a jellegzetes ikonokon keresztül. Néhány szolgáltatás vázlatoszerű felsorolása:

— Logikai döntéssfa vizuális, grafikus reprezentációja (a dBASE-ből ismert VALID függvénynek megfelelő, akár bonyolult felületrendszer is megadható).

— Egyszerűbb grafikus elemek automatikus megrajzolása: téglalap, keret, kör, ellipszis, árnyékvetítés.

— Különböző figyelemfelkeltő betűtípusok, változatos betűméretek.

— A dBASE vagy Paradox adatbázis az OV-ból is közvetlenül megtervezhető és létrehozható.

— Btrieve és ASCII textállományokat is közvetlenül képes beolvasni és kezelni, Sybase/SQL Serverhez csatlakoztatható.

— A DDE (Dynamic Data Exchange) és OLE (Object Linking and Embedding) szoftvertchnológia folytán az OV képes akár hálózatos üzemmódban is együttműködni bármely dBASE vagy Paradox programmal, azokkal közösen,

és egyidejűleg képes adatállományokat karbantartani.

(Egészen extrém esetben elképzelhető a következő is: Windows-os felületen kommunikál programunk a tisztelt felhasználóval, de az adatfeldolgozást egy, a háttérben futó, Clipperben írt program végzi.)

— Az MS Windows révén minden nyomtatott gond nélkül kezel.

— Objektumorientált képernyőtervezés, minden képi elem és adamező könnyedén elmozgatható, áthelyezhető, mérete, színe, betűtípusa, vonalvastagsága egyszerűen megváltoztatható.

A csomag tartalmazza az Object Vision 2.0 royalty free runtime változatát is, ez az a program, amelyet az OV-ban írt alkalmazásainkhoz mellékelhetünk, mivel ez a verzió nem tartalmazza az interaktív tervezői funkciókat. A programozók fantáziájának megmozgatása érdekében még további, mintegy húsz kész, tetszés szerint módosítható alkalmazást is mellékeltek az OV Runtime lemezen.

Ára az Alaplap Postán keresztül 16 800 Ft + ÁFA.

Harvard Graphics for Windows 1.0

Szabvány teremetője

A klasszikusnak mondható Harvard Graphics az amerikai Software Publishing Corporation terméke (más ismert Windowsos termékük a Harvard Draw és a Superbase 2 és 4).

A Harvard Graphics (továbbiakban HG) prezentációs programcsalád, melynek már nemcsak DOS alatt futó, hanem Windowsos implementációja is létezik (ezúttal ezt a változatot ismertetjük behatóbban).

Valódi piaccgazdaságokban egy valamirevaló cég menedzserei rendszeresen, évente több alkalommal is bemutatják a vállalkozás részvényeseinek és igazgatótanácsának az adott üzleti időszak addigi eredményeit és a következő üzleti periódus megvalósítandó tervét. A legszemléletesebben diavetítéssel, számítógépes demonstrációkkal színesítve adják el (ő) mondandójukat a nem felületlenül képzett közgazdászokból álló

hallgatóságának. A lustábbja már csupán számítógépet használ, és abban a hitben ringatja magát, hogy lépést tart a korrál.

A HG a korábbi megszokott eszközök (írásvetítő fólia, diakockák, előre megrajzolt, sokszorosított és kiosztott grafikonok stb.) helyett vagy mellett azok hatásán túltesz: képes önálló számítógépes szemléltetésre is, akár végtelemített (avagy interaktív) slide-show-k megtervezésére és megjelenítésére is.

Persze egy PC-s demonstráció képeit egyenként is ki lehet nyomtatni, sokszorosítás és bekötés után a prezentáció emlékeztetőjét az előadás kezdete előtt pár perccel ki is lehet osztani, amibe a jelenlévők menet közben bejegyezhetik észrevételeiket, kérdéseiket.

A fent leírt technológia minden egyes fázisát támogatja a HG programcsomag, kezdve a bemutató vezérlőnek megírásától a feliratozáson keresztül az egyes grafikonok, szemléltető diagramok automatikus elkészítésén át magáig a szabadon futó prezentáció megjelenítéséig képernyőn — illetve mindezeknek kiíratásáig lézernyomtatott, plotteren, lefényképezéséig instant diákészítővel. A program a következő szolgáltatásokat nyújtja:

— vázlatkészítő (outliner) és szövegszerkesztő;

— angol nyelvű helyesírás-ellenőrző külön egy saját magunk által is szabadon bővíthető és módosítható kivétel-szövegszerkesztővel;

— általános grafikon- és szövegszerkesztő (Chart Gallery): főcímek, alcímek, felsorolások, táblázatok, hierarchikus blokkdiagramok, oszlop-, vonal-, kör-, minimum-maximum, terület-, pont- és szalagdiagramok, vagy az előbbieket tetszés szerinti kombinációja 2 vagy 3 dimenziós effektusokkal, szí-

nekkel, látványos betűtípusokkal feltüpirozva (mintegy 80-féle diagram-típus);

— beépített kis táblázatkezelő a grafikonadatok kényelmes és egyszerű beviteléhez (oszlop- és sorfunkciók, átlagszámítási és egyéb statisztikai algoritmusok);

— slide-show megjelenítő, „diarendező üzemmód” (ScreenShow);

— HyperShow: programozható, interaktív ScreenShow, mely alkalmas szabadonfutó vásári demók, menüvezérelt oktatóprogramok és más szabad formátumú prezentációk megkomponálásához (látványos megoldások: áttűnések, képek be- és kiűztatása, elhalványítása, lepergetése stb.);

— 30 stílusminta, több mint 500 kész sematikus ábra, és rugalmasan változtatható méretű szimbólum (nyílak, keretek, emberkéik, országok sziluetje — köztük Magyarország normál és megyetérképe, valamint színes zászlaja —, piktogramok, szabványos folyamatábra-elemek, stb.);

— a legkorszerűbb szoftvertéchnológiák felhasználásával (DDE = Dynamic Data Exchange és OLE = Object Linking & Embedding) a HG kétirányú képes együttműködni más Windows-alkalmazásokkal (pl. MS Excel, 1-2-3 for Windows);

— beépített objektumorientált rajzolóprogram: bármilyen elforgatható, tükröz-

Minimális hardver- és szoftverkövetelmények

- Bármely PC, melyen fut az MS Windows 3.x standard vagy enhanced üzemmódban (286, 386 vagy 486-os processzorral), de legalább 386-os az ajánlott.
- DOS 3.1 vagy előbbi verzió.
- EGA vagy VGA, utóbbi ajánlott.
- Eger nyomtatékosan ajánlott, de nem szükséges, tablet (mutatótábla) pótolhatja az egeret.
- Minimum 2 Mbájt RAM, de 4 Mbájt az igazi.
- Minimum 8 Mbájt hardiszkkapacitás, de az összes opció installálásához 14 Mbájt szükségesítetik.

Támogatott outputeszközök és szoftverek:

- Az összes Windowson keresztül vezérelhető printer, plotter és direkt dia- vagy filmfelvétel, többek között az Apple, Agfa, Canon, Epson, Hewlett-Packard, IBM, Lasergraphics, NEC, Polaroid, QMS és Tektronix modellek.
- Videokártyák: EGA, VGA, SuperVGA, 8514/A, amelyek legalább 256 kB RAM-mal rendelkeznek.
- Hálózatok: 3Com 3+ Share, Novell ELS, Advanced, SFT és Netware 386, IBM PC

LAN, Banyan Vines 386, Microsoft OS/2 LAN Manager (DOS Client), IBM OS/2 LAN Server (DOS Requester) — ha több felhasználó is akar egyidejűleg dolgozni, úgy 1, 5 vagy 20 felhasználósnál ún. LAN License Packot kell még vásárolni (áruk rendre a teljes értékű csomag árának 0,7, 3 vagy 13-szorosa);

— Fonttechnológiák: Bitstream Speedo (a csomagban benne foglaltatik), Bitstream Facelit, Adobe Type Manager;

— A korábbi DOS-os Harvard Graphics 2.x és 3.0 chart és symbol állományait képes felolvasni, beolvas Lotus 1-2-3, MS Excel és ASCII textállományokat.

Az Alaplap Posta útján többféle áron érhető el:

Harvard Graphics for Windows	1 053 000 + ÁFA
Harvard Graphics I/W Upgrade	19 000 + ÁFA
Harvard Graphics 3.0 for DOS	59 000 + ÁFA
Harvard GeoGraphics for DOS	39 900 + ÁFA
Demólemez: 299 Ft + ÁFA, melynek árát visszavádják az, aki később a teljes értékű szoftvert is megvásárolja az Alaplap Postától.	

hető, torzítható, a méretek fokozatosan állíthatók, színeket, kitöltő mintákat, rasztereket használhatunk;

— strukturált, hierarchikus információkezelés, vázlatkészítés támogatása.

A világon a prezentációs szoftverek közül nem véletlenül a Harvard Graphicsből adták el a legtöbbet, jelenleg már közel két és fél millió darabnál tartanak.

Harvard Draw for Windows 1.0

A CorelDRAW méltó vetélytársa

A prezentációs programok ősatyját, a klasszikusnak mondható Harvard Graphicsot követő Harvard Draw for Windows (a továbbiakban HDW) az amerikai Software Publishing Corporation terméke. A HDW-t mint univerzális rajzolóprogramot elsősorban desktop publishing (DTP) programokhoz ajánlják: ezek kiegészítésére, színesítésére ábrákkal; illetve önálló grafikai műalkotások létrehozására szánják.

A félreértések elkerülése végett: a HDW-nek semmi köze a Bostonban található híres Harvard Egyetemhez. Emez Kaliforniában készült.

Tekintettel arra, hogy nagy lemezfaló program, mely teljes installálás esetén 30 Mbájt szabad helyet tenne magáévá,

valamit tenni kellett. A megoldás: az intelligens üzembe helyezésű program megengedi, hogy akár fájlszintre ma szoláshassunk a kínálathól. Ez különösen áll a tömörített mennyiségű grafikus ábrára és szimbólumra. Mi is csak a legfontosabb szolgáltatásait soroljuk.

— A szabadkézi rajzolás messzemenő támogatása: az egyik ablakban megnyitva dolgozunk a részletre, míg egy másik ablakban teljes színeiben és nagyságában pompázik művünk.

— Bármikor több színpaletából válogathatunk, sőt saját tervezésű színpaletákat is kifejleszthetünk. A színhatásokat tovább fokozhatjuk különböző kitöltőminták alkalmazásával. Egy adott területet akár befoglaló vonalnak megfelelően körkörösen, csillagszerűen, avagy adott színáramlaton keresztül tölthetünk fel (elég megadni a kiindulási és a végső színt).

— A HDW objektumorientált program, így minden egyes részletet külön kezel, a teljes ábrát meglehetősen nehéz óvatlan mozdulatokkal elrontani. Ha

mégis mellényűlnánk, a 16 szintű visszaállító funkció (undo) mindig segít.

— A legtöbb hasznos funkció nemcsak menükön és billentyűkombináción keresztül érhető el, hanem külön ikontömbök (floating dialog boxes) könnyítik meg a program gördülékeny kezelését.

— Autotrace funkció: egy más rendszerről átvett rajz vagy szkennelre beolvasott ábra körvonalait automatikusan kiegyenlíti!

— Online helprendszer.

— Ún. Harvard Draw Script Language, mellyel összetett ábrákat generálhatunk egyetlen tollvonás nélkül, normál ASCII-szöveges formában, mérnöki pontossággal!

— Geometriai alapformák rajzolása: kör, ellipszis, négyszög, sokszög, szabályos ívek, szögek. Ábrák tengelyes és középpontos tükrözése, elforgatása, döntése, torzítása, fokozatmentes zoom.

— Használható mértékegységek: inch, centiméter, pont.

— Főliaszerző egyszerű 99 állású lapot tud kezelni, ezeken tetszés szerinti grafikus objektum szerepelhet.

— Speciális effektsok: két eltérő alakzatot képes egymásba átvinni a közbeeső fokozatok megszerkesztésével, képrészletet megsokszoroz, maszkol (egy képből egy minta szerint vág ki részleteket).

— Szövegek megadott vonalra, mintára illesztése, árnyalása, kitöltése, körvonalazása.

— A csomag ráadásképpen egy műanyag vonalzó és betűmércét is tartalmaz!

A HDW relatív gyorsaságában és hatékonyságában túlnói ki a többi ha-

Minimális hardver- és szoftverkövetelmények

- Bármely PC, melyen fut az MS Windows 3.x standard vagy enhanced üzemmódban (286, 386 vagy 486-os processzorral), de legalább 386-os az igazán megfelelő.
- DOS 3.1 vagy előtti verzió.
- Hercules, EGA vagy VGA monitor; ez utóbbit leginkább, mely képes 256 szín megjelenítésére 640x480-as felbontásban is.
- Egér nyomtatékosan ajánlott, de nem szükséges.
- Minimum 1 Mbájt RAM, de 4 Mbájt a legjobb, melyből 2 Mbájt RAMdrive-t érdemes lefoglalni az átmeneti állományok gyors kezelése érdekében.
- Minimum 10 Mbájt hardisdiskkapacitás, de az összes opció és mintatárolás installálásához plusz 20 Mbájt szükséges.
- Ha nagy bitmap ábrák importját is tervezük, hozzunk létre a Windows SWAPFILE nevű segédprogramjával egy 8 Mbájtos átmeneti állományt, amennyiben 386-os gépeinken enhanced módban kívánjuk futtatni a HDW-t.

Támogatott outputeszközök és szoftverek:

- Az összes, Windowson keresztül vezérelhető printer, plotter és direkt dia-vagy filmfelvétel, többek között az Apple, Agfa, Canon, Epson, Hewlett-Packard, IBM, NEC, Polaroid, QMS és Tektronix modellek.
- Speciális berendezések: General Parametric Videosaw, Agfa Matrix SCODL, Lasergraphics LFR/PFR, PTI Montage FR1.

- Videokártyák: EGA, VGA, SuperVGA, 8514/A, amelyek legalább 256 kb-ji RAM-mal rendelkeznek.
- Hálózatok: 3Com 3+ Share, Novell ELS, Advanced, SFT és Network 386, IBM PC LAN, Banyan Vines 386, Microsoft OS/2 LAN Manager (DOS Client), IBM OS/2 LAN Server 1.3.
- 47 beépített Bitstream skálázható betűcsalád.
- Harvard Graphics 2.x, 3.0 chart és symbol állományait képes beolvasni.
- A beemelhető vektor- és bitmap állományok: AI, PCX, WMF, DRW, BMP, TIFF, CH3, CHT, GCM, SYM, DXF.
- Kimenő (export) állománytípusok: SCODL, AI, TIFF, PCX, BMP, WMF, CH3, PIC, GCM, SY3, DXF, LL, EPS, Windows clipboard.

Az Alaplap Postán keresztül többféle áron érhető el:

Harvard Draw for Windows 1.0	59 000 + ÁFA
Harvard Draw I/W Competitive Upgrade	19 000 + ÁFA

(Konkurens programok — például CorelDRAW, Micrografix Designer, GEM Artline, Adobe Illustrator, Arts & Letters, PC Paintbrush, Aldus Freehand — felhasználói a program eredeti kézikönyvének első oldalát átdávva jogosultak a kedvezményes bevezetési árra.)
Demólemez (1 db 1,2 MB-os 5,25"-es lemez):
299 Ft + ÁFA, melynek árát visszakapja, a ki később a teljes értékű szoftvert is megvásárolja az Alaplap Postától.

sonszóró program közül. Egy szórólapon éppen a CorelDRAW 2.0-val hasonlítják össze, a versengésben a HDW győzött, a mérkőzés végeredménye 21:7 lett.

Javarást sebességben múlta felül vetélytársát, néhol bizony tízszer, tizen-

kétszer is sebesebben oldva meg ugyanazt a tesztfeladatot. Mint tudjuk, az idő pénz. De ha egy program — amiért mi magunk fizettünk — húzza az időnket, és borzolja idegszálainkat, itt az ideje, hogy lecseréljük egy olcsóbb függébre...

E számunk hirdetői

Info#	Oldal
Barex	28 55.
Bürotech	14 21.
Cédrus Rt.	01 B2.
Cédrus Rt.	26 38.
Cédrus Rt.	27 44.
Compmark	20 25.
ComputerLand	03 B4.
Computer Praxis	16 21.
Data Doctor	30 54.
Datentechnik	09 16.
Europrofil	24 37.
Éksóft	12 16.

Info#	Oldal
Floppyland	05 K4.
Galax	21 30.
Hoktrade	29 56.
Huncomp	34 51.
IQ Stúdió	22 37.
IR Szerviz	35 57.
Keszo	04 K4.
Kürt	06 K4.
Macroda	18 22.
Magics	25 37.
Made Info	07 15.

Info#	Oldal
Mézon	02 B3.
Offi-Comp	08 16.
PC Comp	19 22.
Peron	23 37.
Qwerty	11 16.
Sharp	17 60.
Szinva Net	15 21.
Telehold	13 16.
Toner	33 51.
Unitrade	10 16.
Unitrade	31 51.
X-Byte	32 51.

Civilizált Ön?

Játéktér: a jó öreg anyaföld

Hazánkban is sokan játsszák szívesen a különböző szimulációs játékokat. Ezek közül is a legizgalmasabbak a nem csupán egy eszközt (vadászrepülő, harci helikopter vagy tengeraltjáró), hanem egy komplett énszabályozó rendszert modellező programok.

Már az Alaplap hasábjain is megismerkedhettünk a SimCity és SimEarth nevű zseniális játékokkal. (Megjegyezzük, hogy a Maxis — az előbbi két termék kiadója — azóta meglepettette a SimAnt nevű újabb „dilijét” is; itt egy hangyataradalom tevékeny részesevé avat a program). Ezúttal szintén egy olyan programot igyekeztünk bemutatni játékos kedv olvasóinknak, mely térben az egész földet (sőt a játék vége felé a világűr is) felleli. Időben pedig az értelemmel bíró ember megjelenésétől (Krisztus előtt 4000-tól) az első úrkolóniai letelepítéséig fogja át a korokat. A program neve: Civilization!

A Civilizációt a Microprose sikeres szerzője, Sid Meier és csapata készítette. (Korábbi világsszámuk az 1990-ben az év stratégiai játéka címet elnyert Railroad Tycoon volt.) A játékos hanyatló birodalmakat virágoztathat fel logikus döntéseivel. Induláskor választható ugyan más geofizikai és klimatikai paraméterekkel rendelkező bolygó is, de higgyék el, sokkal borzongatóbb úgy játszani, hogy az ellenséges népek földrajzilag ugyanazon a helyen találhatók, ahová őket a világtörténelem predesztinálta. Az ókori görögök a Balkánon, a rómaiak az Appennini-félszigeten, a mongolok Belső-Ázsiában, az aztékok a mai Mexikó területén stb...

Nem a szakszervezeti vezető!

A játék egyúttal társadalom-, gazdaság- és technikatörténeti lecke-ként is felfogható. Több nagy történelmi alakkal is találkozhatunk a szórakoztató időutazás során, így felbukkan Abraham Lincoln, a macedón Nagy Sándor, az angol I. Erzsébet királyné, a mongol világhódító Dzsingisz kán, a babilóniai Hammurabi, a római Julius Caesar, az indiai

Mahatma Gandhi, a kínai Mao Ce-tung, az azték Montezuma, a francia nagy Napóleon, az óegyiptomi Ramszesz, a zulu származású Shaka, az orosz (tudjuk: grúz) Sztálin és II. Vilmos Frigyes, aki nem kevesebb, mint ötven évig volt Poroszország uralkodója.

A játék alapegysége a város. Kezdetben 1, 2 vagy 3 vándorló hordával kezdhetjük a nagy civilizációs kalandot. A hordák alkalmas terepviszonyok és földrajzi fekvés, növényzet, folyó vagy állóvíz közelsége esetén letelepedhetnek, és várost alapíthatnak. A városok kezdetben önellátók. Ha van elegendő élelem, a város lakossága folyamatosan gyarapszik. Megszervezik a városi milíciát, később önálló portyázásra is alkalmas bandériumokat, szabadcsapatokat is toboroznak. A városlakók adót fizetnek nekünk; mi a fővárosban (az elsőnek alapított településen) élvezzük a javukat egyre bővülő királyi palotánkban.

Kenyér és cirkusz

Idővel az emberek közül kinevelhetünk csempüregokat, színészeket és egyéb, a nép szórakoztatását szolgáló művészeket. Ezekből később adószedő tiszteltek, majd pedig tudósokat emelhetünk ki. Utóbbiak meggyorsítják egy város szellemi és anyagi gyarapodását, míg a színészfélék a lakosság jó kedélyállapotát vannak hivatva szolgálni. Mert bizony, ahogy azt a történelemlényekből is megtanulhattuk, az elégedetlen, jóllakatlan ember nem szereti a rendet, lázadozhat, sőt urambocsá', forradalommal le is taszítathat bennünket a trónról.

A városok a technika fejlődésével összhangban egyre ütőkéesebb szereket állíthatnak. Mennél nehezebb és

fejlettebb fegyverzettel küldik őket újukra, annál biztosabb, hogy idegen városokat, népeket hódítanak meg vagy törölnek el a Föld színéről.

A Civilizáció ötféle nehézségi fokozatban játszható: Chieftain (törzsfőnök), Warlord (hadúr), Prince (herceg), King (király) és Emperor (császár). A feladatunk egyre nehezeedik, ahogy a magasabb rangú fokozatot választjuk. A legkönnyebb fokozatban még számos tanácsokat kapunk főtanácsosainktól. A nehezebb fokozatokban már egyre harciasabbak az ellenséges törzsek, népek, és az újabb technikai vívmányok kifejlesztése is több időt vesz igénybe. A kezdetek kezdetén megadhatjuk, hogy hány törzs boldoguljon velünk együtt a földkerekségen, háromtól hétig szabadon kijelölhetjük riválisaink számát. Majd nemzetiséget is választhatunk, s a választásunknak megfelelően azonnal egy ismert személyiség nevét kapjuk kölcsön.

A játék pontra megy. A pontokat a program bonyolultabb összefüggések alapján számolja, s bizony szűkmérűen méri. A kapott pontok száma természetesen korrelál az össznépszerűséggel, a társadalom megelégedettségével, a kiépített infrastruktúrával, a hódítások számával, és a kereskedelem és banktechnika minőségével. Bármikor nyugállományba vonulhatunk, ekkor a program kiértékeli addigi munkánkat. Lehetőség van a mindenkor állást el is menteni lemezre, s a korábbi állást bármikor vissza is tölthetjük. A játéknak nyilvánvalóan akkor is automatikusan vége szakad, ha az egyik nép utolsó városunkat is leigazza, azaz kihalunk, illetve ha az összes népet/nemzetet egyenként meghódítottunk.

Tájak, korok, emberek

Az idő rohamléptekkel telik. Kezdetben 20, majd 10 évet lép egyszerre az idő visszafordíthatatlan kereké, később 2, utóbb egy évet, de ekkor már 15-20 várost kell kormányozni. A helyváltogatás léptéke körülbelül megegyezik Magyarország mostani területével. Az új helyen sokféle növényzet fogadhat bennünket: sarkkőri, sivatagi, hegyi, dzsungel, magas hegységi, óceáni, al-

földi, folyómelléki, mocsaras, tundra vagy erdei. A mezőgazdaság mielőbbi kifejlődésének érdekében természetesen az alföldi és erdei növényzet és talajviszonyok a legkedvezőbbek, de megfelelő öntözési kultúrával (amit ki kell fejlesztenie tudósainknak és feltalálóinknak) még a sivatag is termővé tehető. A következő természeti kincsekből gazdálkodhatunk, ha találunk perzse: szén (a hegyekben), hal (az óceánból), vadak (erdős területen vagy tundrán), nemesfémek, drágakövek (dzsungelben), arany (magashegységekben), lovak (síkságon, alföldön), oázis (sivatagban) és kőolaj (mocsárvidéken).

A bárbar szín: piros

A nagyobb népektől függetlenül bárhol felbukkanhatnak kisebb harcias, bárbar törzsek, amelyeknek egyedüli célja a zsákmányolás és pusztítás. Színük mindig piros. Az ellopott technikai vívmányok nyomán egyre jobb hadifelszereltséggel és kultúrával rendelkeznek, így mind nehezebb lesz őket visszaverni.

Az alapképernyő hat fő részre tagolódik. A legfelső sor a menüléc, innen minden egyes funkció rámutatással is aktivizálható, de a parancsok és beavatkozások zöme egyszerű beütépparancsokkal vagy egérről is kivitelezhető. A bal felső nyolcadban található a kicsinyített sematikus világtérkép. Alatta fényes palotánk kicsinyített mása (melynek pompázata egyenes arányban áll sikerességünkkel, a nép általunk érzett megbecsülésével és hódolatával. Ez alatt látható a státusablak, mely állandóan jelzi nemzetünk összlakosságának számát, az aktuális évszámot, egy villanyórázt, amely a tudományos fejlődést van hivatva jelképezni (mennél világosabb, annál közelebb vannak tudományos munkatársaink a legközelebbi korszakalkotó felfedezéshez. Ugyancsak itt jelzi az ők szín ábra a környezetszennyezés mértékét is, ez inkább az iparosodás korától válik komolyá. A következő sor a birodalmi kincstár tartalmát mutatja, mellette a gazdaság legfontosabb mutatói, a fényűzésre, az adóra és a tudományra fordított összegek aránya. Ez alatt helyezkedik el az aktuális mozgó egységek jellemző azonosító ablak. Információt ad a kérdéses egység nemzetiiségéről, típusáról (katonai, szállító, kereskedő, diplomata, telepes stb...), megtudható a rangja (például veterán, ha már több csatanyerés van a háta mögött), hátralévő lépcsőszáma, az őt anyagiilag finanszírozó város neve, a lefoglalt terület növényzete, valamint látható az ugyanazon a mezőn táborozó

más egységek szimbolikus képe (melyek fekete kerettel a megerősített, „beásott” egységek, árnyaltan az érgységnek kirendelt csapatok, amelyeknek nem mozdulnak el a helyükről, csak ha az őket küldő város visszarendeli őket). Az örök automatikusan teszik a dolgukat, ha ellenséges erő közelíti meg őket.

Egy egységet három fő adat jellemez: a támadó (A = attacking), a védekezési (D = defense) és mozgási (M = movement) faktor. A hadba fogható egységeket a relatíve magas A és D faktor jellemzi.

A tengeren is járhatunk, de csak korszerű tengeri alkalmatosságokkal, úgy mint: gálya, flotta, vitorlás, teherhajó, csatahajó. Az iparosodott korokban már tengerről is levegőből is érheti támadás a városokat. A kisebb városok eltűnnek a a térképről, míg a nagyobbak egyszerűen gazdát cserélnek. Mód van seregek felosztására (disband) és ember alkotja eszközök (hidak, öntözőrendszerek, bányák) szándékos lerombolására.

A telepek üttörő szerepet látnak el az egész játék folyamán. Valójában a tudósokon kívül ők képviselik egyedül a produktív szférát. Termővé teszik a földeket, öntözőműveket hoznak létre, bányákat tárnak fel, mocsarakat csapnak le, erdőtmentényeket és városokat építenek, megszüntetik a környezetszennyezés okozta károkat; utakat, a főbb utak, keresztcsozódások mentén pedig vasutakat építenek (kellő technikai peremfeltételek esetén).

Lám-lám...!

A diplomaták (mint az életben már megszoktuk) nem éppen tisztá játékokat űznek. Egy városba vagy egy ellenséges csapat tagjai közé belopakodva a következő diplomáciai erőfeszítéseket hajthatják végre: egy város erőforrásait kémkedik ki, nagykövetséget alapíthatnak, ipari titkokat deríthetnek föl, azaz ellophatnak idegen technológiát, szabotázs- és terrorakciókat vihetnek véghez (hid-, templomrombolás, a városfalak meggyengítése), forradalmat szíthatnak a kővénél rövidebb, találkoznak a rivalis uralkodóval, végül jó pénzzel meg is vesztegethetnek és rávehetnek átállásra ellenkező oldalon szolgáló egy-

ségeket. Békét köthetnek, áruba bocsátanak technológiákat, hadat üzenhetnek, szövetségeket köthetnek harmadik nemzet rovására.

A kereskedelmet a karavánok alapozzák meg. A kereskedés hoz be az országra luxustárgyakat, javakat, építőanyagot a világ hét (meg még 14) csodájának megépítéséhez. Bármely honi, 10 mezőnél távolabbi város elérése karaván létrehozhat ún. országot, amelyen később vámot szedhet. Idegen kultúrák közötti teremtheti meg a békés kapcsolatot.

A legváltozatosabb palettát az egyes fegyverek, seregtípusok mutatják. A korokban és a korok közötti gatlástalanul csapongva megpróbáljuk felsorolni a lehetőségeket, feltüntetve a fentebb említett A-D-M faktorokat is:

Armor (páncélozott hadosztály, autómobil megléte szükséges):	10-5-2
Artillery (tűzérség, robottechnika kell):	12-2-2
Battleship (hadihajó, acélipar kell):	18-12-4
Bomber (bombázó, fejlett aviatika):	12-1-8
Cannon (ágyú, fémkohászat):	8-1-1
Carrier (repülőgép-anya hajó, fejlett aviatika):	1-12-5
Catapult (katapult, fejlett matematikai ismeretek):	6-1-1
Cavalry (lovasság, lovaglás ismerete):	2-1-1
Chariot (harci szekér, a kerék ismerete):	4-1-2
Cruiser (romboló hadihajó, robbanómotor ismerete):	6-6-6
Fighter (vadászipar, repülés ismerete):	3-3-10
Frigate (fregatt, mágnesség ismerete):	2-2-3
Ironclad (páncélozott monitor, csatahajó, gőzgép ismerete):	4-4-4
Knights (lovagok, a lovagi rendek megjelenése):	4-2-3
Legion (légión, vasmegmunkálás ismerete):	3-1-1
Mechanized Infantry (gépesített gyalogság, szakszervezetek?):	6-6-3
Millite (bandérium):	1-1-1
Musketeers (muskétások, puska megjelése):	2-3-1
Nuclear (atommeghajtású rakéta, rakéte technika és magfűtő):	99-0-16
Phalanx (falang, bronz-megmunkálás):	1-2-1
Riflemen (mesterlövészek, zsoldos katonaság intézménye):	3-5-1
Sail (vitorlás hajó, navigáció ismerete):	1-1-3
Submarine (tengeraltató, tömeggyártás kifejlesztése):	8-2-3
Transport (teherhajó, iparosítás):	0-3-4
Trireme (gályahajó, térfélpéskészítés ismerete):	1-0-3

Az államformák sorrendben a rab-szolgálatotól társadalomtól az anarchián keresztül monarchia, kommunizmus, köztársaság és demokrácia lehetnek. Érdekes módon a programban pontosan ebben a fejlettségi sorrendben alakulhatnak ki! Elgondolkodtató. A mi államformánk: Magyar Köztársaság. Jó úton haladunk a demokrácia irányába.

Sertepertőlő tanácsadók, világcsoádák

Hatféle tanácsadó sertepertőlő körülöt-tünk. Egy városi (belügyi vagy inkább önkormányzati?!), egy katonai, egy hír-szerzési, egy közhangulati, egy keres-kezelmi és egy tudományos tanácsadó. Nem árt megfogadni, amit sügnak, azonban gyakran két tanácsadó egy-másnak ellentmondó instrukciókat ad, ilyenkor nekünk kell döntenünk, melyi-keket fogadjuk meg. Különböző érdekes statisztikákat is kérhetünk: a világ cso-dálataira méltó teljesítmények (Colosseum, nagykönyvtár, kínai nagy fal, függőkertek, világítótorny, jóda, pí-ramisok, Kopernikus csillagvizsgáló-ja, Darwin utazása a Galápagos-szige-tekre, Isaac Newton iskolája, J. S. Bach katedrálisa, Magellán expedíciója, Mi-chelangelo kápolnája, Shakespeare színháza, az Apollo űrprogram, a rák ellenszere, Hoover Dam — az első vízi erőmű —, Manhattan Project: atom-fegyverek építésének terve, SETI pro-gram: földön kívüli értelmes lények fel-kutatását célzó akció, a nők választójog-a(!?), az ENSZ hivatalos helyeinek listája. Lekérhetjük a világ öt legna-gyobb városát, a civilizációban legelő-rehaladottabb nemzetek rangsorát, a vi-lágterképet, az űrhajókat (ha vannak), és egyéb demográfiai tablókat (népes-ség, nemzeti jövedelem, az ország te-rülete, az írní-olvasni tudók aránya, betegségek, környezetszennyezés, vár-ható élethet, családméret, katonai szol-gálat, a gazdaság termelékenységé stb.)

A véletlenül fellépő természeti és egyéb katasztrófák: földrengés, éhínség, tűzvész, árvíz, kalóztámadás, pes-tis, vulkánkitörés bármikor jelentős lé-lekszámvészteséggel fenyegetik váro-sainkat. A programban rendelkezésünkre áll egy egyszerű kislexikon, a Ci-vilopédia, itt minden egyes fogalom, technikai és társadalmi vívmány pontos angol nyelvű leírását megtaláljuk, ki-lőn kiemelve, hogy ez milyen más mutatókat befolyásol, illetve ennek megjelöléséhez melyek a feltétlen kri-tériumok.

A városok egy természetes fejlődé-sen mennek keresztül, de ez a folyamat kincstári segíddettel felgyorsítható.

Előbb építhetünk magtárakat, lakta-nyákat, templomot, piacot, könyvtárat, vízvezeték, kócsokot, bírósági épüle-tet, stadiont, hajókat, a város köré védőfalakat, gyárt, palotát, erőművet, újrafeldolgozó üzemet (recycling cen-ter), egyetemet, űrvédelmi eszközöket (SDI — Strategic Defense Initiative), a világ csodáinak bármelyikét, előbb to-borozhatunk seregeket, telepéseket, mint ahogy azok maguktól kialakulná-nak. Persze a város el is adhat vagy bérbé adhat épületeket, ha plusz bevételre van szüksége.

A kőművesség és a magfűző

Az érdeklődés kedvéért még néhány fontosabb lépést is felsorolunk, mely az emberiség életében mérföldkönek tekinthető: a kőművesség, az írás, az asztronómia, a filozófia, a puskapor, a vegyészet, a robbanóanyagok, az elekt-ronika, a komputer, a génebézészet, a műanyagok, a szupravezető anyagok, a nukleáris és magfűzős energia felfede-zése.

Ugye, magas fűd szellemi repülésre biztatjuk azokat, akiket megfogott eme nagyszerű játék ötletessége?

Ugye, már rájöttek a címben feltett kérdés értelmére? Mert aki még nem civilizált (a Civilizációval), az nagyot mulasztott. Próbálja ki Ön is! (Még a földtörvény meghozatala előtt!)

A program (Sid Meier's Civilization, Microprose) ára az Alaplap Postában: 6400 Ft + ÁFA.



A Maxis Software sokat adott, mégis adós

Tavaly már egy ízben részletesen ismertettük a SimCity nevű város-szimulátort (lásd 91/5., 37. old.). Ez-úttal a program Microsoft Win-dowsos változatára szeretnénk fel-hívni a figyelmet. Előrebocsátjuk, hogy a játékszabályok semmiben sem változtak, minden maradt a ré-giben, azonban az összes manipu-láció legalább ötször kényelmesebb lett.

Minden beavatkozási lehetőség csinos kis ikonok kapott, így egérrel azonnal elérhető. Még a PC szerény képességű, egycsatornás hangszó-róján is monumentálisan szól zenei alapmotívumot kapott a program, va-lamint a bejelentkező képernyőn Maxine, a téhen mosolyog kedélye-sen az ábrázatunkba. Az új installa-ció eljárás képes a SimCity Gra-phics 1 és 2 „városdizsziertező” pro-gramok beolvasására is, így a hagy-mányos építőelemek helyett azonos

szabályok mellett építhetünk az ókori Ázsiában, a középkori Európá-ban vagy a vadnyugaton, illetve a 2000-es évek USA-jában, 2050 Európájában vagy éppen egy kolónián a Holdon. Megjegyezzük, hogy ez a program már nem tartalmaz védelmet, így nem kell megvárni a feltört változatát sem.

Mindenképpen érdemes áttérni rá, azaz „polgármester kerestetik”!

(Erről eszünkbe juthat — tudva, hogy a Maxis Software elkészítette a nagy sikerű SimEarth bolygószimulátorának a Windowsos változa-tát is —, hogy más is kerestetik! A hangyaboly-szimulátor — SimAnt — mellett kíváncsian várjuk a Robo-Sport for Windows nevű új dilit, amely transzformerek és egyéb, nem emberszabású géplények vég-telen változatosságú harcaival bor-olja — vagy nyugtatja? — ide-geinket.)

AKCIÓ!

Megrendeléstől a kulcsrakész átadásáig: **72 óra!**



IRODA KULTÚRA STÚDIO

1067 Bp. VI., Podmaniczky u. 27.
Tel.: 131-8188, Tel./Fax: 132-0188
Bemutatóterem:
1054 Bp. V., Kálmán I. u. 14. Tel./Fax: 153-4898
Videóiroda:
Pécs, Szalay A. u. 12/A Tel/Fax: (72) 21-181

EZT ZSEBRE TEHETI...



A PC 3100-as palmtop = IBM PC + menedzszerkaikulátor

- 640x200 LCD display
- 2 MB RAM
- 1 MB ROM
- RS 232C és Centronics interface
- 77 gombos billentyűzet
- DOS 3.3
- Laplink szoftver
- CPU: 80C88A/10 MHz
- Opciók:
 - 2 db PCMCIA 1.0 IC memóriakártya
 - 1 db 3.5"/1.44 MB FDD
 - Áramellátás: 3 db ceruzacell (LR 6)

A készülék súlya mindössze 480 gramm!

Fogyasztói ára: 90.000,- Ft + ÁFA, REKLÁMÁRON!



Kapható:

EURO-PROFIL

1147 Budapest XIV., Fűrés u. 65/b
Tel.: 163-5210, 163-6095



...gyors
emelkedés

PERON · REKLÁM Kft.

Telefon: 149-4819



ICS

FESTÉKSZALAG

és

CÍMKE

minden Magyarországon működő
VONALKÓDNYOMTATÓHOZ.

Rendkívüli ajánlat: minden második

I N G Y E N

van!

Mag ICS

Informatikai
Rendszerfejlesztő és
Marketing Kft.

11-9400 Sopron, Bátya u. 75. Tel./Fax: (36) 99/14-250



incas®



**Bármilyen festékkazettát keres,
nálunk megtalálja!**

CÉDRUS KAROLINA ÁRUHÁZ
Budapest XI., Karolina út 17. • Tel.: 166-2111 • Fax: 185-2221

Prix Ars Electronica 1992, Linz

Átlépés egy másik dimenzióba

A rendezvényről hazatérve azzal összegeztük élményeinket, hogy ez egy óriási kiállítás volt. No nem a méreteit, hanem a bemutatott alkotásokat tekintve. A művészi teljesítményhez pedig kitűnő háttérrel nyújtott az, hogy az egyik fő szponzor a Silicon Graphics, amely manapság a legjobban használható RISC-processzoros grafikai munkahelyeket gyártja.

Ezt az egyidejűleg több helyszínen futó linzi kiállítást és rendezvénysorozatot 1987 óta évenként megrendezik, jelentős összegű díjakkal jutalmazva a legjobbakat. Ebben az évben például 1,25 millió schillinget osztottak ki, megtoldva az erkölcsi siker értékével. Csak egyetlen példa: 1987-ben a zenei díjat Peter Gabriel, a Genesis egykori alapítója és énekese kapta. Hazai és magyar

származású külföldi alkotók is szép sikereket értek már el az elmúlt években.

A verseny jelenleg az alábbi 4 kategóriában zajlik:

- Számítógépes grafika
- Számítógépes animáció
- Interaktív művészetek
- Számítógépes zene



Dennis Muren—Mark Dippé—Steve Williams:
Terminátor II

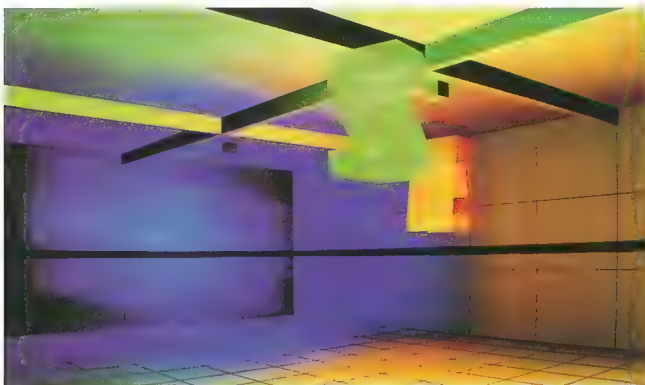
Virtuális (látszólagos) valóság

Számomra az idei rendezvény egyik fő érdekessége a VR, az úgynevezett „virtual reality” volt. Képzljük el, hogy kap az ember egy furcsa szemüveget, amelyben színes LCD képernyő van. Ezen láthatók a számítógép által közvetített képek. Ehhez eredetileg még egy fejhallgatót is kapunk (bár a kiállításon ezt már lespórolták), amin a virtuális világ hangjait halhatjuk. Végezetül a jobb kezünkre ráhúznak egy kesztyűt, amely érzékeli kezünk és ujjaink mozgását, s amellyel megadhatjuk, hogy merre akarunk haladni.

Ennyi technikai előfeltétellel tehát „beöltözik” az ember a „virtuális valóságba”. Három dimenzióban élhetünk át egy olyan világot, amely elszakad hétköznapi valóságunk dimenzióitól. Ahogy forgatom a fejem, leguggolok, felegyenesedek, úgy változik a képzetbeli táj nézőpontja, mintha csak valódi lenne. Ahogy mozgatom a kesztyűs kezem, arra haladok, amerre az ujjam mutat, gyorsíthatok, lassíthatok, megállhatok, visszafordulhatok. Benézhetek például egy kúp alá, amely a föld felett öt centire lebeg, láthatok mindenféle furcsa spirális képződményt, óriási emberi fejet, természetesen fejjel lefelé... meg mindent, amit az alkotói



William Latham: *Fénysugárkövetés változatok a végletekig repülőgépén*



Jeffrey Show: Virtuális Múzeum — 5. terem

fantázia kitalált. A valós világgal szembeni különbség eleinte ijesztő, de meg lehet szokni... utána pedig az ember alig talál vissza.

Lássuk azért a dolog technikai oldalát is! A két szem számára természetesen külön-külön képet kell létrehozni, real-time módon. Az óriási számítási és képfeldolgozási sebességet két külön Iris Crimson (Silicon Graphics) biztosítja. Egyik az egyik szemre és fülre dolgozik, a másik pedig a másikra. A kettőt normál számítógéphálózaton keresztül egy Macintosh köti össze. Van továbbá az érzékelőkről bejövő jeleket feldolgozó, és a számítógép számára megfelelő formátumúra hozó három fekete doboz, s azok rettenetes kábelzsungelen keresztül vannak összekötve a Mac-kel. Végezetül egy óriási

projektor is tartozik a mutatványhoz, hogy a kívülállók is élvezhessék ezt a háromdimenziós utópiát.

Virtuális múzeum

Továbbmenve láthatunk az előzőhöz hasonló elvű „múzeumot”. Egy forgó korong áll egy üres terem közepén. A korong egyik szélén fotel, szemben vele pedig egy kivetítő, alatta a sokat sejtető Silicon Graphics 4D/310 VGX szupergéppel.

Ha az ember beül a fotelba, és elkezd oldalra fordulni, az egész forgókorong vele fordul, a képernyőn megjelenő kép pedig, (amely egyébként valódi műzumi tárgy mása) úgy változik, mintha tényleg elfordultunk volna egy valódi teremben. Ha a fotelban előre dőlünk,

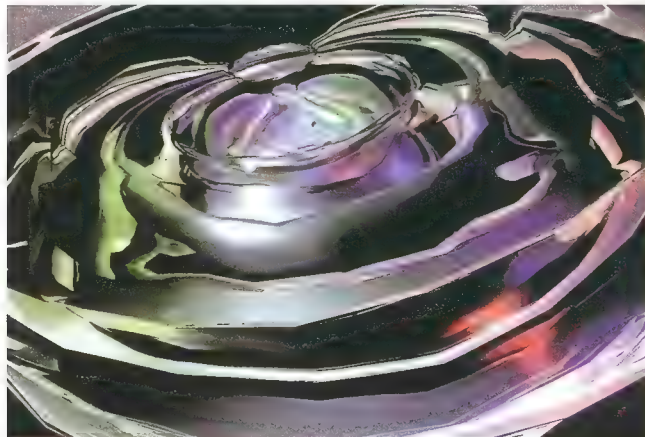
ezzel a múzeumban előre megyünk, ha hátra dőlünk, akkor pedig hátra. Ha elég közel vagyunk egy ajtóhoz, előre dőlvé átsétálhatunk a másik terembe. Az igazi látványosság mégis az, amikor meglátjuk az egyes termek „életét”. Az egyikben óriási világító számok ugrándoznak a padló és a plafon között, a másikon mindenféle kínai frászelek jönnek ki a falból az egyik oldalán, s tűnnek el a másikon. Az egész olyan érzetet kelt, mint annak idején a Beatles együttes „Sárga tengeralttjáró” című rajzfilmje.

„Szemforgatás”

Szintén nagyon érdekes és magyar vonatkozású virtuális világ volt Agnes Hegedűs „Handsight” című alkotása.



Andrew Wilkin—Michael Kass: Reakció-diffúziós gombfelületek. (Részlet)



Stewart McSherry: Chevrolet

Ez az installáció kevesebb helyet foglalt el, mint az előzőek, leszámítva a szoba sarkában meghúzódó Silicon Graphics Power Series 4D/310VGX gépet. Az egész tulajdonképpen egy biliárdgolyó nagyságú „szem”, amit kézbe fogva egy kb. 40 cm átmérőjű, felül nyitott üveggömbben lehet vele „nézelődni”, azaz a „szemmel” belenyúlni a gömbbe, és ott „szemet forgatni”. Az üveggömb a valóságban természetesen üres, de a kivetített képen úgy látszik, mintha egy halszem-optikával nézve sokféle tárgyat — szerszámot, rugót, harapófogót, kalapácsot — vennénk szemügyre. Úgy változik a kép, ahogy a kezünkkel mozgatjuk a gömbben a „szemet”.

Egészen furcsa alkalmazást mutatott be egy német szerzőpáros. A falon egy tévéképernyő volt, képkerettel bekere-

tevez, a képernyőn pedig egy németalföldi reneszánsz mester képe, és „első ránézésre” az egész úgy nézett ki, mintha igazi festményt látnánk. A trükk akkor működött, ha leültünk a képpel szemben, és elkezdtünk nézegetni a „festményt”. A kép ugyanis furcsa módon mindig ott masztolódott el, ahová éppen néztünk. De honnan tudhatja egy számítógép, hogy a kép melyik részét vesszük szemügyre? A megoldás a kép alatt elhelyezett, félig áteresztő tükör, amely mögött egy infra kamera a széket ülve a képet figyelő ember szemére volt ráállítva. Egy kis kalibrálás után a kameráról lejövő kép, amely az ember szemmozgását mutatja, „könnyen” konvertálható koordinátákká, annak alapján pedig már igazán semmiségi valamilyen módon megváltoztatni a képet azon a részen, ahová a szemlélő néz.

Filmeffektusok

Szóljunk egy olyan műről is, amelyet nem állítottak ki, azonban mint referenciát a kiállítás összes anyaga említett teszt róla. Ez pedig Dennis Mure, Mark Dippé és Steve Williams alkotása James Cameron filmjében, a Terminátor II-ben. (Amikor ezt a cikket írom, az MTV-n éppen Arnold Schwarzenegger veszi át a legjobb férfi szereplőnek járó díjat a fenti filmben nyújtott alakításáért).

Aki látta a filmet, biztos, hogy nem felejtii el, amikor a T1000-es átfolyik a börtönrácson, vagy amikor „felkel” a kockaköves folyosóról és felveszi az ór alakját. Ezeket az effektusokat természetesen szintén a Silicon Graphics gépeivel hozták létre. Pontosabban szólvá harmincán is több Iris 4D grafikus munkaállomás dolgozott a film képeinek létrehozásán, s azokon futott a rengeteg szoftver, többek között az Alias 2.

A klasszikus műfaj

A Prix Ars Electronica pályázat és kiállítás egyik hagyományos ágazata a számítógépes grafika. Talán ezen mérhető le legjobban a számítástechnika művészeti eszközként való felhasználásának fejlődése. Az első időszakban ugyanis sem az animáció, sem a zene nem volt jellemző a számítástechnikai alkalmazásokra, legalábbis nem a nagyközönség számára elérhetően. A grafika viszont kezdeti fogva élenként foglalkoztatta a számítástechnikusok és a művészek fantáziáját. Viszonylag könnyen hozzáférhető gépekkel is tet-

szetős grafikái műveket lehetett alkotni, legfeljebb a számítási idő volt hosszabb, a felbontás és a színek száma pedig kevesebb.

Az Ars Electronicán Andrew Witkin és Michael Kass nagyon szíves, valószínűleg grafikákat mutatott be, természetesen Silicon Graphics gépen, saját fejlesztéssel szoftverrel készítve.

Stewart McSherry kiállítása anyaga viszont azt mutatja, hogy az alkotó a tükröződő és áttetsző felületek ábrázolásában éli ki magát. Ehhez a Silicon Graphics 4D/210 gépen futó Alias 3.0 raytracer (fényugárkövető) szoftverre is szüksége volt a felületek fénytörési és fényvisszaverési tulajdonságainak visszaadásához.

De nem minden grafikához szükséges feltétlenül a Silicon Graphics valamelyik számítógépe. Itt van például Bullás József és Krajcsik Zolt grafika, amely bonyolult textúrájú felületeket ábrázol, és Compaq 486-os gépen készült. Persze egy kis grafikai bővítés kellett hozzá, hogy a felbontás és az árnyalatok számát megfelelően lehessen bővíteni. Ezen a képen is látszik, hogy elkészítése nem két percig tartott.

Egy másik „PC-alapú” grafika Yoshiyuki Abe absztrakt, tükröződő felületeket ábrázoló alkotása, amely talán az egyik legletisztultabb, és szintén 486-os gépen készült, fényugárkövetéssel.

William Latham viszont inkább az IBM 3090-re esküszik, de valószínűleg azért, mert „ez volt kéznél”. Ilyen gépet senki nem tart otthon a lakásban, kivéve talán az örült milliommokat. (Nem mintha a Silicon Graphics nagyobb gépei elérhetőek volna a magánember számára.) Ez az IBM 6090-es képernyőn bemutatott grafika egészen élet-hű színlábú szörnyet ábrázol, de az sem baj, ha valaki mára asszociál.

Mitől mozog?

Az óriási „mozgó” látványosságokat itt nem tudjuk illusztrálni, pedig azok között is volt néhány figyelemreméltó alkotás. Például Karl Sims animációs produkciója, amelyhez (szerényen) egy Connection Machine System CM-2-es gépet használt. Erről a gépről tudni kell, hogy minimum 65536 (!) processzort tartalmaz, s azokat sejtprocesszorként lehet programozni, Lisp programozási nyelven. Ezek után elképzelhető a gép számítási sebessége. Ami viszont nem képzelhető el, az az ára. Nem is „futkosnak” ezerszáma a világon ezek a gépek, és Karl Sims sem otthon játszadozott, hanem munkája során a Massa-

chusetts Institute of Technology futott össze vele. Az említett animáció készítésekor egy kiinduló képre az Artificial Evolution programmal ráengedett számítási algoritmus a képből valamilyen „mutáns” hozott létre. Kissé módosított algoritmus azután mindig újabb mutáns keletkezett. Most képzéljük el az egészt úgy, hogy minden egyes képponthez egy processzort rendelünk, hadd foglalkozzon külön vele. Az újabb mutációk létrehozására tehát nem kell sokat várunk. Amikor a művészek valamelyik megtesztzik, azt veszi kiinduló képnek, mintegy beavatkozva az „evolúció” menetébe. A végeredmény egy 2 perc 15 másodperces, valamint egy 1 perc 35 másodperces animáció. Egyszerű, nem?

Illúziókeltés és -rombolás

Mint minden hasonló rendezvényen, az az Ars Electronicán is voltak szegényes, anakronisztikus, és oda nem illő alkotások, de nem ez volt a jellemző, s azokkal nem is foglalkozom. Szóljunk inkább egy-két érdekes installációról.

Tetszett például Christian Möller „Space Balance” című alkotása, egy sötét teremben lévő libikóka, amely — ha az ember közepén ráállt — ide-oda billegett. Az egészesnek a lényege viszont az, hogy a terem két szemben lévő falára vetített kép a libikóka meghosszabbítását mutatta, mégpedig úgy, mintha az egész terem velünk együtt mozogna. A teremben még kisebb-nagyobb golyók is voltak, amelyek szintén össze-vissza gurultak, sőt hangot is adtak, amikor visszapattantak a falról.

Igen érdekes volt az a kiállítás is, ahol régi idők analóg szintetizátorait mutatták be, például a Moog és ARP gépeket. Ezeket ki is lehetett próbálni, sőt természetesen hatalmas hangzavar volt az egész múzeumban.

Végül néhány szót két egymást követő „performance”-ról. Az egyiknek „IMmediaCY” volt a címe. Összintén szólvá szörnyű volt. Laurie Andersont akarta utánozni, kevés sikerrel. Már a címére sem emlékszem viszont a másiknak, ahol MIDI-vezérelt hegedűből csíhítottak elő rettenetes hangokat, és ezeket össze-vissza képekkel szinkronizálták. Akinek tetszett, lelke (szeme?) fűlt? rajta!

Összegezés helyett csak annyit tudok mondani, hogy jövőre is kimegyek az Ars Electronicára. Ha valahol, akkor itt képet kaphatok arról, hogy hol tart a számítógépes-elektronikus művészet.

Sík Zoltán

Az MPIR páciens-nyilvántartó információs rendszer

Organikus, permanens folyamat

Az alább vázolt elvek alapján a szerzőknek sikerült körülbelül másfél évi folyamatos fejlesztéssel egy hasznos, stabilan működő, számos vonatkozásban előremutató, „egészséges” rendszert kidolgozniuk.

A cikket különösen ajánljuk a medicina területén dolgozó, rendelőkben, kórházakban foglalkoztatott olvasóinknak, de a magánpraxist folytató doktoroknak is — mellettük azonban mindazoknak, akik szakértő (jellegű) rendszerek iránt érdeklődnek.

Az olyan bonyolultságú fejlesztések, mint amilyen például egy, az orvosi munkát támogatni kívánó számítógépes rendszer, nem végezhető a hagyományos, skolisztikus módszerekkel — azaz egy felmérés-analízis munkafázist követő, „végleges, teljes, kimerevített” programrendszer kidolgozásával. Ugyanis:

- dinamikusnak változnak a feltételek, bővíthetnek az alkalmazható eszközök (hardver, szoftver);
- gyakorta módosulnak az ügyviteli, elszámolási eljárások, előírások;
- az adott fejlesztés realizálása nyomán törvényszerűen alakul az alkalmazásról, a rendszerrel kialakított kép, pontosabb lesz a feladat- és célrendszer;
- a nehezen, illetve csak részben strukturálható, de meghatározó, döntő fontosságú szakmai, orvosi ismeretek, munkamódszerek (diagnosztizálás, terápiabehatárolás, kiértékelés stb.) számítógépes támogatási lehetőségei is permanensen változnak, finomodnak a fejlesztésekkel. (A fejlesztési zsákutcából sokszor többet lehet tanulni, profitálni, mint a sikerekből.)

Mi ilyen fejlesztési eszközöket (jórészt saját CASE-jellegű bővíthetősékek) használunk, amelyekkel a gyakorlatban a változásokkal, a felvetett észrevételekkel, igényekkel szinkronban végezhető a rendszerfejlesztés anélkül, hogy közben a felhasználónál valamiféle hátrány (adatvesztés, időbeli kiesés) jelentkezne. Nem „végleges”, előzetesen „agyonlemezett” megoldásra törekszünk, hanem egy működő, és a gya-

korlatban gyorsan verifikálható és módosítható megoldás kialakítására. Eközben szorgalmazzuk a minél szélesebb körű alkalmazásbavételt, és a felhasználók észrevételeinek azonnali visszacsatolását a fejlesztésbe.

Amikor a rendszert valamilyen fórumon bemutatjuk, a hazai szoftverfejlesztői szakma képviselői általában megkérdezik, hogy milyen fejlesztőeszközzel dolgozunk. A választ halálra pedig, hogy az MPIR moduláris felépítésű rendszert a Clipper relációs adatbázis-kezelőre alapozva dolgoztuk ki, mindig felvetődik:

- Melyik a legkorszerűbb, a leghatékonyabb eszköz?
- Miért éppen ezzel fejlesztünk, miért nem egy újjal dolgozunk?

A mi „lovunk”

Az MPIR rendszer fejlesztése hét évvel ezelőtt kezdődött el a dBase/Clipperre alapozva. Akkoriban PC-n ez volt a „legkorszerűbb” eszköz. Az egyre összetettebb feladatok megoldásán keresztül alakult ki az a saját fejlesztői programcsomag, valamint jött létre az az ismeretbázis, amellyel a mai feltételek mellett is hatékonyan tudunk dolgozni. Természetesen jól tudjuk, hogy az újabb, fejlesztést támogató eszközökkel kezdő szinten is gyorsan és megbízhatóan lehet rugalmas nyilvántartási rendszereket létrehozni, és mi magunk is intenzíven foglalkozunk — már munkakörünkön kívül fogva is — az újabb és legújabb eszközök alkalmazásbavételével. Az az „üzemszerű” állapot, hogy a fejlődéssel meg kell tanulni

együtt élni, együtt dolgozni. A cselekvést, a konkrét fejlesztői munkát azonban nem pótolja semmi, és a bevált szerszámot a legtöbb esetben nem érdemes cserélni.

Szakosodás/szakértés

Anélkül, hogy itt részletesebben belemennénk a szakértő rendszerek igen sok nyitott kérdést tartalmazó problematikájába, néhány átfogó megállapítást kívánunk tenni.

A modern értelemben vett számítástechnika alkalmazásának kezdetein már kidolgoztak olyan programokat, amelyek a fizikai, műszaki rendszermodellezés területén az ún. optimális megoldások, rendszerállapotok és beavatkozások behatárolását sikeresen oldották meg. Azonban ezeket az analitikus, numerikus, klasszikus értelemben vett matematikai eljárásokat már az összetettebb fizikai, műszaki rendszereknél sem lehetett alkalmazni, hanem sajátos heurisztikus keretbe ágyazott „kevert”, egyedi problémafüggő programok kerültek kidolgozásra (szimulációs, sajátos problémafüggő/elemező, kereső eljárások).

Az összetett, rosszul strukturált, dinamikus gazdasági, társadalmi, biológiai, orvosi stb. rendszereknél, feladatoknál pedig gyorsan túllépte az idő a misztifikált „automatikus, optimális” számítógépes fejlesztési törekvéseket. Helyettük a számítógépes „támogatás, segítség”, valamint a komplex, bonyolult feladatoknál a „szakértő rendszerek” fogalma és ezekkel kapcsolatos irányok léptek.

Meglátásunk szerint a realizált újabb szakértői rendszerek mindig valamilyen relációs adatbázis-kezelőre épülnek — figyelembe véve a jelentős és bonyolult adatfeldolgozási igényeket —, az alapvető feladatot azonban az adott, többnyire „élő”, nehezen strukturálható („fuzzy”) adat-, ill. ismeretmodell, valamint a feldolgozás módja jelenti. A problémák modellezése döntően heurisztikus, alkalmazva a komplex döntések területén feltárt eljárásokat, azonban ezeket sem szolgálnak. Elsősorban mint megközelítési mód dominál ez: kreatívan, nagyfokú adaptívi-

tással, kihasználva az adott feladat sajátosságaihoz szarmazó lehetőségeket, és figyelemmel a rendszert használóakra.

Az MPiR rendszer döntéstámogató moduljai is a leírtak szellemében készülték.

A rendszer három fő alkalmazói modulcsoportból áll, amelyek fejlesztését és forgalmazását is különböző stratégia szerint alakítottuk.

Az alpmódulrendszer ingyenes, szabadon másolható. Ennek ellenére adatnyilvántartási szempontból korlátozások nélkül használható. Tartalmazza az alapadat-nyilvántartást és a páciensekhez kapcsolódó kezelések nyilvántartó és lekérdező eljárásait, valamint az adat- és rendszerbiztonsági modulokat. A tapasztalatok szerint ez mint alpmódulrendszer önmagában is jól használható.

A felhasználó a rendszer ún. regisztrálásakor számos, saját adataival specifikált modulhoz jut, s ezenkívül külön megveheti a számára szükséges egyéb bővíthetéseket.

Az MPiR a fentiekben kívül tartalmaz komplex, expert jellegű alkalmazói modulokat is: döntéstámogató, szakértői (géppel támogatott diagnosztizálás, terápiaajánlás stb.) eljárásokat is, amelyek jelenleg gyakorlati tesztelés alatt vannak (magánpraxisban, oktatásban, kutatásban). E cikkekben azonban csak a főbb részek általános leírására szorítkozunk.

Szeretnénk itt megjegyezni, hogy mi csak a lehetőségeit teremtettük meg annak, hogy a program PC-s környezetben, a gyakorlati munkában is képes legyen érdemi segítséget nyújtani. Ami a továbbiakban is alapvető, és a gyakorlatban hatalmas és folyamatos munkát jelent, az az ún. ismeret- és tudásbázis bevitelle a rendszerbe. Ez még kiegészítésre, pontosításra szorul, és szívesen vennék olyan orvosok, orvoscsoportok jelentkezését, akik részt vennének a speciális modulok kidolgozásában, pontosításában, tesztelésében.

Kartonok modul

A betegek adatainak rögzítésére szolgál, gyakorlatilag a kartotékrendszer váltja ki. Sokféle szempont szerint támogatja a gyors visszakeresést. Minden karterhoz kapcsolható kb. száz A4-es oldalnyi szöveges információ. A több szempontú logikai keresést ezekben a leírások is erősen támogatják. Továbbá lehetőség van arra is, hogy a páciensek tetszőleges számú és megnevezésű logikai csoportokba sorolhatók legyenek. Az így besorolt betegek közül bármelyik listánál válogathatunk: „csak

azok szerepelnek rajtuk, akikre igaz hogy...”. Ugyanezzel a válogatási lehetőséggel tehetjük meg azt is, hogy a kartonok közül csak azokat másolja ki a rendszer lemezre, amelyeket meghatároztunk. Ez az adatmentési lehetőség tulajdonképpen egy postázó modul. A beteg összes adatait kérésre kimásolja lemezre, és így az hordozható. Átadható másik egészségügyi intézménynek, vagy akár hatóságnak.

Kezelések modul

Ebbe a táblába, mint egy pénztárgépbe, rögzíthetők azok az adatok, amelyek alapján a hatóságok felé generálhatók a szükséges statisztikák és listák, a legkülönbözőbb bontásban és sorrendben.

Betegségek modul

A BNO-kódok listája mellett még számos hasznos adatot tartalmaz, továbbá a betegségek rövid algoritmikus leírását is.

Vizsgálatok modul

A betegekben végzett vizsgálatokat és azok eredményeit tárolja itt a rendszer. Előválogatott csoportok teszik kényelmessé és gyorsá az orvos munkáját.

Kinyomathatók a labornak szóló levelek, de — ha hálózaton működik a rendszer — nem az orvosnak kell az eredményeket befűia, hanem azt a labor munkatársai végzik el, és a következők az osztályon. A labor dolgozó nem tudja név szerint, kinek a vizsgálatairól van szó. Ezzel támogatni kívánjuk az orvosi titoktartás érvényesíthetőségét, és a személyi jogok védelmét. Normálérték-szótár is tartozik ehhez a táblához, mely azonnali segítség ad az adatok kiértékeléséhez és viszonyításához. Gombnyomásra grafikusan, idősrba rendezetten jelennek meg a beteg vizsgálati adatai a képernyőn, így az orvos szemléletesen láthatja visszamenőleg is az egyes értékek alakulását.

Diagnosztizálást támogató modul

Ez a modul már több helyen váltott ki érdeklődéssel párosuló vitát. Nem véletlenül, hiszen szerte a világon folynak fejlesztések és kutatások ezen a területen, és ószintén szólna tartotunk egy kicsit az orvosi szakma kedvezően hozzáállásától. Különösen a hasonló témát kutató szakemberek óvatk benünket a gyakorlatban dolgozó kollé-

gáiktól. De furcsa (vagy talán nem is furcsa) módon éppen a gyakorlati orvosok részéről tapasztaltunk olyan aktív érdeklődést és támogatást, amely nem csak inspiráló és erőt ad, hanem meghatározó módon hasznosítható volt. Erre azonban még sokáig szükségünk is lesz — itt magunk nem boldogulunk.

Tréning modul

A program egy képzéleleleli beteget generál az alábbi módon:

1. Vélelengenerátor segítségével egy konkrét betegségre utaló tünetsorozatot állít elő, normál adatokkal vegyesen.

2. A felhasználó eldöntheti, vagy a géptől kérheti azt, hogy az adott értékek magasak, alacsonyok vagy normálisak-e.

3. Meg kell határozni a valószínűsíthető betegségeket.

4. A programrendszer a benne lévő tudásbázis alapján magyarázatokkal egyetemben felsorolja a szóba jöhető betegségeket, csökkenő valószínűséggel, illetve bizonytalansági tényezőkkkel.

5. A felhasználó meghatározhatja az általa alkalmazandó terápiát.

6. A programrendszer is megadja a javasolt gyógymódot.

Ez a modul az oktatásban, a kutatásban, a rendszer tesztelésében, ellenőrzésében és a benne lévő ismeret/tudásbázis finomítására alkalmazható.

Gyógymódok modul

Itt tárolhatók az alkalmazandó gyógymódok. Értjük ezen az európai, a keleti, vagy akár a természetgyógyászatban alkalmazottakat is, korlátozások nélkül. A gyógymódajánló modul képes arra, hogy ezek közül előválogassa az orvos számára azon gyógymódokat, amelyek az adott betegség(ek)nél szóba kerülhetnek, de továbbra is az orvosnak kell ezeket bevizsgálnia, értelmeznie, majd dönteni. Mindenesetre a rendszer felhívja a figyelmet arra, hogy

- mely gyógyszereket szed most a beteg, milyen gyógyszerkezelés(ek) alatt áll;

- mely gyógyszerekre, gyógyszer-csoportokra érzékeny, melyek ellenjavalltak;

- melyek állnak egymással kölcsönhatásban;

- van-e köztük olyan, amelynek alkalmazásavételéhez további vizsgálatok szükségesek (természetesen, ha elkészülnek a kérdéses vizsgálatok, akkor ellenőrizi azokat);

• tájékoztatást ad az adott gyógymód szokásos adagjairól;

• elolvashatóak a rövid információk (mint például a „zöld könyvben”).

Még számos információ található a gyógymódokról, így például az ár, a felfrásával és alkalmazásbavételével kapcsolatos előírások és szabályok, a hatáskódja, kiszorítási stb. Ezen információkat természetesen nem mi, a fejlesztők tápláljuk a rendszerbe, hanem megfelelő információkkal rendelkező cégek, intézmények, például a HungaroPharma. Meggyőződésünk, hogy az adatokat azoknak kell karbantartani, akik azokkal elsődlegesen kapcsolatban állnak.

A rendszer képes a vények szabályos kiállítására és nyomtatására is.

Határidőnapló

A határidőnapló is hasznos szolgálatot tehet a programcsomagot használó számára. Abban a pillanatban, amikor egy beteget berendel az orvos, egy elfoglaltsági tábla jelenik meg a képernyőn. Itt láthatók a még szabad időpontok, és nem kell fellapozni a naptárat. A program ezt automatikusan vezeti, de felfrható ide az ügyelet, a szabadság, a továbbképzések, és tulajdonképpen mindenféle időbeli elfoglaltság.

Kórházi, ill. többfelhasználós környezetben lehetőség van arra, hogy

közös, szabad időpontokat keressünk, pl. egy megbeszélés időpontjának kijelöléséhez.

Eszközők

Ez egy egyszerűsített készletnyilvántartó modul, melyben nyomon követhető az egyes tételek mozgásai; például hogy ki, mikor, kinek és mennyiért adott vagy vett át valamit, hogy mennyi a pillanatnyi készlet stb.

A futtatási környezet

Minimál konfiguráció — egygépes felhasználásra:

1db IBM XT/AT-kompatibilis számítógép:

- központi tár: 640 kb-át,
- 1 db floppyegység,
- 1 db merevlemez egység,
- monitor: monokróm vagy színes,
- néhány floppy az adatok mentéséhez,
- DOS 3.3 operációs rendszer.

Az MPIR rendszer működik többgépes, hálózatos környezetben is, ezáltal alkalmas körzeti rendelők, kórházi osztályok munkájának komplex támogatására is.

A rendszerben a felhasználó a megszokott szakmai, alkalmazói kifejezéseket használhatja.

A menüvezérléssel és a help szolgáltatással önmagát tanítja, magyarázza a rendszer. (A helpet a felhasználó átruhajta, bővítheti.)

A többszintű jogosultság (ún. password-hozzáférés), az egyszerűen vezérelhető rendszerkarbantartási funkciók az adat- és rendszerbiztonságot szolgálják. A korszerű adatbázis-kezelés és fejlesztő eljárások biztosítják a rugalmas bővíthetőséget, az egyszerű módosítást, a továbbfejleszthetőséget.

Végezetül

(és némiképp célzatosan)

A rendszerben található egy message-box is. A rendszerben dolgozó felhasználók üzenhetnek egymásnak még akkor is, ha a címzett nincs éppen a helyén. Majd megkapja, amikor legközelebb „belép”. Még egyfelhasználós környezetben sem használatlan, hiszen a délutáni váltás üzenhet a délutáninak. A mi munkahelyünkön azonban, ahol fut már néhány hálózatos alkalmazásunk — az „SLIST” többnyire 35-40 működő szervert mutat —, a felhasználók előszeretettel használják mindennapi munkájukban a message-box szolgáltatásait. Ezen keresztül kapjuk mi is az észrevételeiket. Üzeneteket azonban másképpen és máshogyan is várunk...

Kiss János — Kereszturi János



Nyárvégi szoftverakció a

CÉDRUS KAROLINA ÁRUHÁZBAN!



Ízelítőt a kínálatból:

Aldus Page Maker 4.0 (3.0 + 4.0 upgrade)	78 000,- helyett	74 000,-
Artline 2.0	49 900,- helyett	35 000,-
MS Windows 3.0 magyar kezelői feldíjtellet	24 900,- helyett	19 600,-
(Az árak a 25%-os ÁFA-t nem tartalmazzák.)		

Szoftverbemutatók a

CÉDRUS KAROLINA ÁRUHÁZ BEMUTATÓTERMÉBEN!

Szeptembertől minden második héten szoftverbemutatót tartunk.

Az első bemutató időpontja: 1992. szeptember 8., 15 óra.

Témája: Adobe Illustrator 4.0 (for Windows)

Részel: Adobe Type Manager 2.0
Adobe Type Align
Adobe Streamline

A bemutató napján

10% kedvezményt adunk az Illustratorra.



Forrasztópáka helyett alkatrészcsere

A hardver(es) jövője

El vagyok keseredve.

Mint hardverrel is foglalkozó szakembernek, megvan rá az okom, amint azt az alábbiakban elmondom. Lehet, hogy nem mindenki ért velem egyet, de egy kicsit talán ők is együttéreznek velem.

Aki megteheti, nézzen bele egy „rég”i XT vagy PC, majd egy 486-os gép belsejébe. Az alaplapot szemügyre véve látható, hogy az XT-ben szereplő áramkörök mindegyike az elektronikai tervezésben általánosan használt, jól ismert ún. katalógus-áramkör (mivel katalógusokban szerepel). A szerelés is a megszokott, nyomtatott áramkörös megoldású. Ez azt jelenti, hogy az integrált áramkörök, kondenzátorok és ellenállások kivezetéseit a nyomtatott áramkört lapon fűrt lyukakon át dugva és beforsztva rögzítik.

Egy 286/386/486-os alaplap már egy más világ: igen finom rajzolatú, több-rétegű nyomtatott áramkört lemezen kialakított, felület szereléses technológiával készülnek ezek, és speciális felhasználóorientált áramköröket tartalmaznak. A felület szereléses technológia teljes mértékben és viszonylag könnyen automatizálható: a hasábszerden kiképzett, a két végén forrasztható kivezetés nélkül gyártott elemeket a nyomtatott áramkört lapon helyezik el, és az adott helyen forrasztással (esetleg ragasztással) rögzítik.

A készülékekhez adott dokumentáció is más. Már nem szerepelnek kapcsolási rajzok, inkább csak blokkvázlatok, valamint az egyéb egységekhez csatlakozó pontok és bekötések elnevezései. Ennek az az oka, hogy az alkalmazott áramkörök nagy része már nem katalógus-áramkör, hanem egy-egy erre szakosodott cég (például: Chips and Technologies, Headland) terméke.

Forrasztópáka és XT eladó...

Halad a világ. Nem figyelték meg, hogy egy tévészerelő már nem annyira meg szokott vendég? A televíziógyártásban alkalmazott elektronikai technológia — hasonlóan a számítógépgyártáshoz — igen nagy megbízhatóságot ad. Nagyon

népszerűek az OSD (on screen display) technológiájú készülékek, amelyeknél a kezelés egy-két nyomógombbal vagy távszabályzóval elintézhető, és a funkciókat a képernyőn jelenít meg. Ez nemcsak divat, hanem annak a felismerésnek az alkalmazása, hogy a mechanikus kezelőgombok, szabályozók könnyebben elromlanak. A legtöbb elektromechanikai rendszer azonban igen megbízhatóan működik.

Jelenleg egy elektronikus berendezés erkölcsi elavulása előbb következik be, mint annak fizikai tönkremenetele. Azt hiszem, most már érthető a szomorúságom. Egyre a forrasztópákás korszaknak, a hibakereséssel, majd boldog reparálással megnyerhető sikereknek. Hiszen hogyan lehet egy felület szerelt technológiával készült berendezést javítani? Nagyon nehezen. Az igen kis méretű integrált áramköröket, ellenállásokat, kondenzátorokat sok esetben elektronsugárral forrasztják, tűhegyű páka használata nehézkes. Minimális kézművességgel is zárlatot okozhatunk az IC-k lábain az oscilloszkóp mérőszondájával. Meg aztán mit mérjünk? Kapcsolási rajz nincs, és még ha meg is találunk a hibás alkatrészt, hogyan és mennyiért szerezzük be azt?

Ezért a javítások nagy része jelenleg kártyacsere. Tessék mérlegelni: a két soros és egy párhuzamos portot tartalmazó kártya ára átfalva együtt 1250 Ft. Érdemes egy ilyen kártyát javítani? Természetesen nem.

Ez az imponáló fejlődés mégis pesszimiztikus jövőképet (is) fest elém. Eddig azért volt sok mérmőkre, szakemberre szükség, mert a termelő berendezésekben alkalmazott technológia működtetése, javítása legtöbbször magas szintű, speciális szakértelmet igényelt. Az új berendezések ritkán meghibásodó részegységeinek cseréje már nem igényel ilyen tudást.

Jó példa az autók elektronizációja. A klasszikus autók elektromos berendezéseit az autovillamossági műszerész szakmunkás javította. Mit tesz ma egy ilyen szakember, ha valaki azzal vízi hozzá speciális nyugati kocsiát, hogy tönkrement az elektronikus gyújtás vagy más elektronikus egység? Egyet tehet: szétbontja a karját, és olyan szervizbe küldi a tulajdonost, ahol speciális műszerek segítségével állapítható meg a hiba, de a javítás legtöbbször ilyenkor is a hibás egység cseréje.

Jobb sem félni, sem megijedni!

Ahhoz, hogy a legújabb bejelentésű chipből komplett, működő PC legyen, csupán memóriára, billentyűzetre és egy monitorra van szükség. A tok maga statikus működésű és kis fogyasztású, ezért ideális a zseb-PC számítógépekhez. A bejelentést úgy időzítették, hogy az egybeesett az IBM-PC tízéves jubileumával.

Már készülnek a szintén egy tokba integrált PC-k fejlettebb, 80x86-os processzort tartalmazó típusának a piaca dobására. A jövő ebből a nézőpontból nagyon izgalmas. Az áramkörök szintű szemléletet és ismereteket egyre szűkebb területen lehet alkalmazni. Valószínű, hogy a mérnöki fejlesztő tevékenységet fel fogja váltani a rendszerek részegységekből összeállított, rendszerintegráló tevékenység. A biztos út a szakértelem átmentésére: a rendszerben való gondolkodás.

Dinamikus és statikus

A nagybonyolultságú digitális áramkörök — mint például a mikroprocesszorok, kalkulátor-áramkörök — belső működésükhez ütemező jelet igényelnek, az órajelet. Ez az órajel biztosítja azt is, hogy az áramkör belső tárolói frissítéssel (folyamatos újrafárasztással) megőrizzék tartalmukat. Ha ez az órajel megszűnik, a tok nem képes helyesen működni. Azt mondjuk, hogy a tok dinamikus működési.

Az újabb fejlesztések eredményeként már gyártanak olyan mikroprocesszorokat (ezek a 80x86-os típusok), amelyeket leállítva — azaz órajelüket megszüntetve — majd újraindítva helyesen működnek. Az ilyen tokokat statikus működésűeknek nevezünk. A statikus működésnek köszönhetően álló állapotban a processzor fogyasztása igen alacsony, ezért a hordozható számítógépekben elsődlegesen az ilyen típusokat alkalmazzák.

Kónya László

Párhuzamosok, mégis találkoznak II.

Taszkkommunikációs eszközök

A párhuzamos programoknak, illetve futtatási környezetüknek — tágabban: a multitaszking rendszernek — a biztonsága és életképessége érdekében különböző kommunikációs protollokat és átkapcsolási mechanizmusokat állítottak csatasorba. Ezeket mutatjuk be Modula-2 platformon.

A taszkkommunikációs eszközök nagymértékben elősegítik a logikai hibák elkerülését a program tervezése folyamán. Ezek az eszközök tulajdonképpen adatcsere-protollok, valamint processzátkapcsolást vezérlő mechanizmusok (lásd a keretes részt).

Általuk biztosíthatjuk, hogy a párhuzamos program különböző processzei között meglegyen az összhang, és a program korrektül, a logikai tervezésnek megfelelően fusson; ne kerüljön deadlock állapotba, minden processz a szerepének megfelelő módon működjön, valamint hogy a processzek közötti adatforgalom logikailag hibátlan legyen.

Dijkstra és a szemaforok

Az absztrakt kommunikációs protollok kifejlesztésének első nagy eredménye Dijkstra nevéhez fűződik. Ő volt az, aki a szemaforok mechanizmusát bevezette a multiprogramozásba. A szemaforokat könnyű implementálni, és elég erős eszközök ahhoz, hogy velük sok taszkkommunikációs problémát elegánsan megoldjunk. A szemaforok lehetővé teszik azt is, hogy erősebb és strukturáltabb kommunikációs eszközöket tervezzünk és implementáljunk.

Definíció szerint a szemafor egy egész típusú változó, amely csak pozitív értékeket vehet fel. A szemaforok két

csoportha oszthatók. Amelyek csak két értéket (0 és 1) vehetnek fel, a bináris szemaforok; amelyek akármilyen pozitív egész számot felvehetnek, az általános szemaforok. Miután megadtuk a szemafor kezdőértékét, csak két művelet hajtható végre rajta. Az egyik a wait(sz) (várj sz-re), a másik a signal(sz) (küldj sz-t). A műveletek a következők:

wait(sz : Szemafor) → Ha sz > 0, akkor sz:=sz-1, ellenkező esetben annak a processznek a végrehajtása, amely meghívta a wait műveletet, felfüggesztésre kerül.

signal(sz : Szemafor) → Ha van olyan processz, melynek a végrehajtása megszakadt egy korábban kiadott wait(sz) hatására, akkor a vezérlés átkerül ehhez a processzhez, ellenkező esetben sz:=sz+1.

Csak a wait és a signal műveletek engedhetők meg a szemaforváltózkodón; különösen tilos a szemaforváltózkodó értékének direkt írása vagy értékének vizsgálata. A wait és a signal műveletek alapműveletek ebben a szerkezetben, tehát végrehajtásuk egymást kizáró fo-

Multiprogramozást támogató eszközök Modula-2-ben

A program az algoritmus leírása, a processz az algoritmus végrehajtása. A program elindításának pillanatában létrejön a főprocessz, amely később maga is hozhat létre újabb processzeket. Ezek a processzek egymással párhuzamosan hajthatók végre. Ha a Modula rendszer egy processzoros gépre van telepítve, akkor egyszerre csak egy processz futhat, és a többinek várakoznia kell addig, amíg a vezérlés sora rájuk nem kerül. A szabvány Modula-ban a következő utasítások állnak rendelkezésre ahhoz, hogy új processzeket hozzunk létre, és a processzek között átadjuk a vezérlést.

NEWPROCESS (proc:PROC; cím:ADDRESS; méret:CARDINAL; VAR task:PROCESS)

Ez az eljárás létrehoz egy processzt, amely a paraméter nélküli proc eljárásból áll. A processznek szüksége van egy munkaterületre, amelynek címét a cím paraméter, méretét pedig a méret paraméter adja meg. Ha a munkaterület túl kicsi, akkor a program futásideji hibával leáll. TopSpeed Modula-ban a munkaterületnek minimum 1000 bajtnak kell lennie. Ha sikerült az új processz létrehozása, akkor a task változóban kapjuk vissza azt a kezelőt, amellyel ezután hivatkozhatunk a processzre.

TRANSFER (VAR mostani, következő:PROCESS)

Ha át akarjuk adni a vezérlést egy másik processznek, akkor ezt az eljárást kell meghívunk. Az eljárás meghívása után az éppen futó processz azonosítója bekerül a mostani paraméterbe, és a processz végrehajtása felfüggesztődik, a vezérlés a következő paraméter által azonosított processzre kerül.

IOTRANSFER (VAR mostani, következő:PROCESS; vektor: CARDINAL)

A processzek közti átkapcsolás történhet tudatosan egy másik processz kezdeményezésére, de történhet véletlenül is, amikor egy bizonyos megszakítás hatására kapja meg a vezérlést egy bizonyos processz. Ahhoz, hogy olyan processzünk legyen, amely kezeli a megszakítást, először a NEWPROCESS eljárással létre kell hozni, majd egy TRANSFER utasítással át kell neki adni a vezérlést. A processz ezután végrehajthat néhány műveletet, majd ciklusba kerül. A ciklusban az első dolog az, hogy meghívja az IOTRANSFER eljárást. Ez az eljárás hasonlóan működik, mint a TRANSFER, de definiál egy megszakító vektort is. Ha az adott megszakítás létrejön, akkor az éppen futó processz felfüggesztésre kerül, és az IOTRANSFER-t meghívó processz kapja meg a vezérlést.

Azért, hogy elkerüljük a megszakítás üjből generálását, amíg a rendszer a megszakításkézeltől hajtja végre, a modulnak, amely a kezelő processzt deklarálta, prioritást kell adni. Ezt a prioritást közvetlenül a modul azonosítója után, szögletes zárójelben adhatjuk meg. A prioritás egy CARDINAL típusú szám, amely miután BITSET típusú konvertálódik, logikai összeget képez a megszakítást kezelő áramkör megszakító maszkjával. Ez a mechanizmus azt eredményezi, hogy amíg a program a modul által deklarált eljárásokat hajtja végre, addig a megszakításvezérlő megszakító maszkja azt az értéket veszti fel, amely az eredeti érték és a modul prioritásának logikai összege. E mechanizmus által tehát leíthatunk különböző hardver-megszakításokat.

lyamat. Tilos egyszerre több processzben ezeket a műveleteket egyszerre végrehajtani. A signal művelet nem köti ki, hogy melyik processzt kell reaktíválni, ha több processz vár ugyanarra a szemaforra.

Ezért a szemafor implementációjától függ, hogy milyen sorrendben kapják meg a vezérlést azok a processzek, amelyek ugyanarra a szemaforra várnak.

Az OOP és a monitorok

Az olyan operációs rendszer, amely csak szemaforokat használ, nagyon sérülékeny: elég, ha csak egy szemafor-művelet van rossz helyen, vagy egyáltalán hiányzik valahonnan, máris az egész logikai rendszer felborulhat. Szükséges tehát egy olyan eszköz — a monitor —, amely saját maga ügyel rá, hogy a kritikus területre egyszerre csak egy processz férjen hozzá. A monitorok ötlete két szálon érett be.

Az egyik bejáratot gyakorlat: az operációs rendszerek funkció-végrehajtó magjára. A legtöbb ilyen mag egy nagy program, amely minden operációsrendszer-függvényt kontrollál, és csak ő hajthatja végre. Hogyha a P1 processz üzenetet akar küldeni a P2 processznek, akkor az üzenetet elküldi az M monitornak azzal az utasítással, hogy ez az üzenet a P2 processzhez tartozik. A monitor elintézi az üzenet átadását. Fő funkciója még az erőforrások feletti abszolút felügyelet. M működését nem lehet megszakítani, ami biztosítja a kölcsönös kizárás elvének betartását. Csak a monitor tud hozzáférni különböző speciális memóriaterületekhez, valamint csak ő tud végrehajtani bizonyos műveleteket, például az I/O műveleteket. (Azok a monitorok, amelyekkel mi foglalkozunk, az operációsrendszer-monitorok decentralizált verziói lesznek. Ezekből lehet egyszerre több is a rendszerben, és mindegyik egy bizonyos funkciót lát el.)

A másik tényező az adatok és a hozzájuk tartozó műveletek strukturalizációjára való törekvés. Ebből nőtt ki a 90-es évek programtechnikai forradalma, az objektumorientált programozási stílus. A monitor tehát lehet (és meg is valósítható mint) egy objektum, amelyben az adatok és az adatokhoz tartozó műveletek együttesére igaz, hogy csak a monitor tud hozzáférni a monitoron belül definiált adatokhoz, továbbá a monitor és a külvilág kapcsolatát csakis a monitor publikus procedúráinak paraméterein keresztül zajlik.

Ha nem állnak rendelkezésre objektumorientált eszközök, akkor a monitor megvalósítható különálló könyvtár formájában, amely csak a külvilággal kapcsolatos teremtő procedúráit „nyitja ki”, és az adatokat, valamint a privát eljárásokat csak az implementációs részben tartalmazza. A monitorokat könnyű implementálni a szemaforok segítségével, és ez lehetővé teszi, hogy a monitorban levő eljárások tényleg megszakítás nélkül futhassanak, egyszerre csak egy processz hívására — ez elengedhetetlen a kölcsönös kizárás elvének betarthatóságához.

Ada és a randevű

A szemaforok és a monitorok centralizált szinkronizációs eszközök, amelyek nem tartoznak egyetlen processzhez sem. Saját maguk szabályozzák a processzek közti átkapcsolás sorrendjét és rendszerét. Ez a szerkezet nagyon jól megfelel egy olyan rendszerben, amelyben csak egy számítógép van, de mi a helyzet az osztott rendszerek esetében? (Azokat a rendszereket nevezzük osztott rendszereknek, amelyek több egymástól független számítógépen futnak, és a rendszer részei közötti egyetlen kapcsolat kizárólag üzenetek cseréjén alapul.) Ez viszont a következő problémát veti fel: az üzenetek kiürülése és felvétele között nincs semmi szinkronizáció, az üzenetek elveszhetnek útközben, és a rendszert alkotó számítógépek közül is meghibásodhat néhány példány. Ilyen helyzetben egy olyan önszabályzó protokollra van szükség, ahol a processz saját maga dönt arról, hogy vár-e (és meddig) az üzenetre. Ha a processz úgy dönt, hogy vár, akkor saját maga függeszti fel a működését, és saját maga dönt arról, hogy meddig fog várni. (A szemaforok és a monitorok esetében más volt a rendszer: ahhoz, hogy egy processz reaktívizálódjon, egy másik processznek ki kellett küldenie egy szignált.) Egy ilyen protokollnak egyetlen fontos eleme a pro-

cesszek közötti kommunikációs csatornák biztosítása, és az üzenetek kicserélése e csatornák felhasználásával. Ezt az üzenetváltást nevezzük randevűnek.

A randevűt Hoare ajánlotta 1978-ban „Cooperating sequential processes” című munkájában. Az eredeti verzió helyett azonban inkább az Ada nyelvben realizálról ejtünk néhány szót. (Az Adáról lapunkban már több írás jelent meg. — A szerk.) A nyelv születésekor úgy hitték, hogy megalkotják a végső nyelvet, amely véget vet az ún. szoftverkrízisnek. (Ez az a jelenség, miszerint a szoftver fejlesztési idejének csak egy egészen kis részét teszi ki magának a rendszernek a tervezése és implementálása, az idő túlnyomó részét a rendszer tesztelése teszi ki. Ha a rendszer meghibásodik, akkor a hibajavításra megint rengeteg időt kell szánni.) Később kiderült, hogy az Ada sem old meg minden problémát...

Az Adában a taszk egy objektumhoz hasonló típus. A randevűt három nyelvi elem biztosítja. Az első egy speciális típus, az ENTRY, amely hasonlít a procedúradeklarációhoz abból a szempontból, hogy meg kell adni az input és output paramétereket. E paraméterek kellenek az adatforgalomhoz a randevű lebonyolító két taszk között. A következő nyelvi elem a SELECT szerkezet, amellyel a taszk egy randevűkérélmel ad ki egy másik taszk ENTRY típusú belépési pontjára, és ha létrejön a randevű, akkor a SELECT szerkezetben definiált változók segítségével megy az adatsere a taszok között. A harmadik szerkezet az ACCEPT: az adott taszk ezzel jelzi, hogy kész elfogadni egy randevűhívási kérelmet egy adott ENTRY típusú belépési pontra. Mind a SELECT, mind a ACCEPT szerkezetben definiálhatunk alternatív műveleteket, amelyekkel akkor hajt végre a taszk, ha valamiért nincs randevű. (Akit komolyabban érdekel az Ada nyelv és a taszkrandevű, olvassa el I. C. Plyde „Az Ada programozási nyelv” című könyvét.)

A TopSpeed Modula-2 multitask rendszere

Az alapkönyvtárak között van a multitask programozást segítő könyvtár is. Ez a könyvtár a következőket biztosítja számunkra:

- általános szemafor típus, valamint a szemaforokon végezhető műveletek;
- egy dinamikus, egyenlő időszelvény időosztáson alapuló átkapcsoló mechanizmus;
- maximum 32 egyszerre futó processz;
- maximum 65 534 szintű prioritáshierarchia;
- futásidőjű prioritásszint-változtatás;
- direkt taszkatkapcsolási lehetőség;
- kritikus területek kezdetének és végének jelzésére szolgáló lock/unlock funkció;
- teljes körű együttműködés a Windows könyvtárral.

A lemez mellékleten van egy könyvtár, amely az Ada taszkrendevű funkcióját implementálja bizonyos megkötésekkel, Modula-2-ben. A könyvtár néhány lehetőségét demonstrálja egy példaprogram, amely egy klasszikus multiprogramozási feladatot old meg a randevű segítségével. Ez az ún. „öt filozófus probléma”, melyben a multiprogramozás szinte minden buktatója benne van.

Öt filozófus egész nap nem csinál mást, mint új filozófiai elméleteket gyárt, és mindegyik örülne, ha csak a gondolkodással kellene foglalkoznia. De időnként megéheznek, és ezért átmennek a konyhába, ahol egy nagy közös asztalnál mindenkinek van egy tányérja és egy villája. Az asztal közepén egy soha ki nem ürülő spagettis táln van. Sajnos a filozófusok csak két villával tudnak étkezni, ezért ha valame-

lyük enni akar, akkor a saját villáján kívül szüksége van a hozzá legközelebb eső másik villára is. Tehát egyszerre csak kettő lakhat jól az asztalnál, és a többinek várakoznia kell, amíg nem lesz szabad villa. Egyezkedni muszáj... (A probléma lényege tulajdonképpen nem más, mint az erőforrások egyenlő, igazságos elosztása az igénylők között.)

Dobi Sándor

A C nyelv Unix-felülete III.

Olvasás innen-onnan, így és amúgy

Nemcsak írni, olvasni is tudni kell! A múlt hónapban volt szó a kíró utasításokról — szóval a felét már megtanulhattuk a dolgoknak —, most jön a bonyolultabb rész...

A `scanf`, `fscanf`, `sscanf` utasítások és a read elsődleges bemutatása után a melyekben rejő tudnivalókról is szó lesz, és ennek kapcsán más fontos fogalmakról, technikákról adunk ismertetést.

Az utasításhármas

A `scanf` utasítással a standard bemenetről, míg az `fscanf` utasítással egy fájlpointeren keresztül fájlból, az `sscanf` utasítással pedig memóriából lehet ASCII formátumú adatokat olvasni és átkonvertálni.

Az utasítások szintaxisa a következőképpen fest:

```
scanf (format[, pointer,
pointer...]);
char *format;
fscanf (fp, format[, pointer,
[pointer...]);
FILE *fp;
char *format;
sscanf (s, format[, pointer,
[pointer...]);
char *s, *format;
```

A 'format' vezérlőstring a legegyszerűbb esetben csak konverziós specifikációt tartalmaz. Egy konverziós specifikáció egy %-jelet követő valamilyen karakterből áll. A megfelelő argumentum (egy mutató) azt adja meg, hogy hová kell konvertálás után betölteni az adatot.

A leggyakoribb típusok a következők:

```
d (decimális egész)
e, f, g (fix vagy
lebegőpontos float)
s (string)
```

A bemeneten a fehér karaktereknek (szóköz, tabulátor, új sor) delimiter szerepük van, a formátumstringben specifikált mezők elválasztására szolgálnak. (Kivétel a későbbiekben ismertetendő karakter- és karakterosztálytípus.) A stringeket (%) is úgy kell értelmezni, hogy fehér karakterek által határolt karaktersorozatok. String olvasásakor a memóriába bekerül a lezáró 0 bájtnál. Például:

```
fp = fopen("DATA", "r");
fscanf(fp, "%d%d", &x, &y);
printf("x=%d y=%d\n", x, y);
fscanf(fp, "%f%e", &f1,
&f2, &buff);
printf("f1=%f f2=%f
buff=%s\n", f1, f2, &buff);
```

DATA tartalma:

```
11 22
1.234E05 5.678
ABCD
```

Kimenet:

```
x=11 y=22
f1=123400.000000
f2=5.678000 buff=ABCD
```

Gyakori hiba kezdőknél, hogy az argumentummegadásnál mutató helyett magát a változót adják meg (ez rossz!):

```
fscanf(fp, "%f", f1);
```

A karakter- és karakterosztály-típus kivételével (ezekről később lesz szó) a konverzióspecifikációk között elhelyezett szóközlőknek nincs jelentősége:

```
fscanf("%f%f", &f1, &f2) =
= fscanf("%f %f", &f1, &f2)
```

Puffer nélkül

A read a blokkolt, puffereletlen olvasás előírása. Fájldeszkriptorral megnyitott fájlból a read utasítással lehet (puffereletlenül) adott számú bájtot olvasni:

```
read (fd, buf, nbytes);
int fd;
char *buf;
unsigned int nbytes;
```

Itt `fd` a megnyitott fájlra mutató deszkriptor, `buf` a memóriapuffer címe, ahová a beolvasott karakterek kerülnek, `nbytes` a beolvasandó bájtok száma. A read visszatérő értéke a ténylegesen beolvasott bájtok számát adja meg sikeres művelet esetén, különben —1. (Ha például `fd` nem egy nyitott fájlra mutat, vagy ha a hozzáférési jogok miatt nem olvasható a fájl.) Például:

```
fd = open("DATA", 0);
n = read(fd, buff, 2);
printf("n=%d buff[0]=
%c buff[1]=
%c\n", n, buff[0], buff[1]);
n = read(fd, buff, 2);
printf("n=%d buff[0]=%c
buff[1]=%c\n", n, buff[0],
buff[1]);
```

DATA tartalma:

ABC

(3 karakter volt a fájlban, tehát a végén nem volt újsor-karakter!)

Kimenet:

```
n=2 buff[0]=A buff[1]=B
n=1 buff[0]=C buff[1]=B
```

A write a blokkolt, puffereletlen írás kezdeményezése. Fájldeszkriptorral megnyitott fájlba a write utasítással lehet (puffereletlenül) adott számú bájt-ot írni:

```
write(fd,buf,nbytes); int fd;
char *buf;
unsigned int nbytes;
```

Itt fd a megnyitott fájlra mutató deszkriptor, buf a memóriapuffer címe, ahonnan a karaktereket ki kell másolni, nbytes a kiírandó bájtok száma. A write visszatérő értéke a kiírt bájtok számát adja meg sikeres művelet esetén, különben —1. (Ha például fd nem egy nyitott fájlra mutat, vagy ha a hozzáférési jogok miatt nem írható a fájl.)

Például:

```
fd = open("DATA",1);
strcpy(buff,"12345");
write(fd,buf,3);
```

DATA tartalma a kiírás előtt:

abcdefg

DATA tartalma a kiírás után:

123defg

Nemes Mihály

Ért Ön Snobolul?

Ne kezdjük újra!

Mostani számunk mágneslemez mellékletén olyan nyelv fordító-értelmező programját adjuk közre, amely után sokáig hiába ácsingóztak nagy európai számítóközpontok munkatársai is. Kérjük olvasóinkat, hogy jól őrizzék meg; lehetőleg készítsenek egy külön könyvtárat, mert szükségük lehet rá az elkövetkezendő hónapokban. Cikksorozatot indítunk a Snobol nyelv megismertetésére, és a mágneslemez mellékleten programozási fogásokat mutatunk be, hogy kedvet csináljunk a nyelvben rejlő lehetőségek kipróbálására.

Tudomásom szerint Európában hosszú ideig egyedül a hollandiai Matematikai és Számítástudományi Központ birtokában volt belőle egy példány, mások csak könyvekből értesülhettek a nyelv (és a felhasználását biztosító fordító-értelmező program) imponáló megoldásairól.

Többek között azokból a tanulmányokból és kutatási jelentésekből is, amelyeket éppen a szóban forgó amszterdami központ munkatársai publikáltak a nyelv látszólagos egyszerűségéről és rejtett bonyolultságáról, valamint a felbukkanó új gondolatok továbbfejlesztési lehetőségeiről.

Miért nincs, ha van?

Valódi unikum, hogy egy ilyen nagy hírnév programrendszer teljes verziója, ha méretkorlátozásokkal is, de a programozók népes táborának közkincsévé

válhatott (lásd a 92/6. szám Közkincs rovátát).

A nyelv a Bell Laboratóriumból indult el kontinenshódító útjára. (Amerikában a legtöbb nagy számítóközpontban már jó ideje hozzá lehetett férni valamelyik változatához.) Különösen az gyorsította meg elterjedését, hogy R. E. Griswold, a nyelv egyik megteremtője — aki később a C nyelv szakértőjeként vált közismertté — kifejlesztett egy önmagában is figyelemre méltó, speciális gépfüggetlen makroimplementáló nyelvet. Az ún. SIL makrónyelv (Snobol Implementation Language) nagybán megkönnyítette az implementáció elkészítését a legkülönbözőbb géptípusokra, és egyben biztosította az így készült szoftvereszköz hatékonyságát is.

Az első működő Snobol rendszerek IBM gépekre készültek, ezeket követte a Burroughs, az RCA, a CDC, a UNI-

VAC, az SDS, majd a General Electric és a PDP — mint látható, lényegében valamennyi nagy amerikai számítógépgyártó cég gépeire készült fordító-értelmező. Mivel azonban ezeknek a kifejlesztését legtöbbször nem maguk a gyártó cégek finanszírozták, így terjesztésében sem voltak érdekelték. Talán elsősorban ez magyarázza, hogy Európába alig szivárgott át. Nem tartozott bele a gyártó cégek standard kínálatába, és beszerzését is csak vonakodva vállalták, horrorbilis összegekért.

A nyelv egyik ismert alkalmazása arra keres megoldást, hogy milyen módszerekkel lehet közönséges szövegekből összegyűjteni, és formalizált alakra hozni a bennük rejlő hasznos információkat. Tudjuk, hogy bizonyos adatbázisok készítőinek a legnagyobb gondot a hatalmas mennyiségű és állandó mozgásban, változásban lévő kémiai információtömeg jelenti. Nos, elsősorban az ő dolgukat könnyíti az ún. SIE rendszer (Specialized Information Extraction System). Ez a szakmai információkivonatoló rendszer morfológiai, szintaktikai, valamint lexicológiai és szemantikai vizsgálatok alapján „nyers” angol nyelvű leírásokból tudja előállítani különböző kémiai reakciók tömör értelmező kivonatát.

Makrók — mikróban

A 80-as évek végén néhány bösz matematikusnak az az istentől elrugaszkodott gondolata támadt, hogy a makróimplementáló nyelv hatékonyságát megpróbálja mikroszámítógépeken is kamatoztatni. A kísérlet a várakozásnál is fényesebben sikerült. Kitűnően oldották meg a számítógépes rendszer memóriagazdálkodását; munka közben automatikusan folyik a háttérben a „személygyűjtés”, a felszabadult hulladékmemória újrahasznosítása, a felhasználónak erre a legcsekélyebb figyelmet sem kell fordítania.

Az is hozzájárult a sikerhez, hogy előzőleg már sokszorosan tesztelték a nagygépeken, így az implementálóknak csak arra kellett kényszerülniük, hogy a PC-kre készült változat az utolsó csavarig pontosan megfeleljen a nagygépeken futó mintarendszerek előírásainak. És tegyük hozzá, ezekre jártak a PC-s minirendszerek készítőinek, hogy a hatékonyság már távolról sem kell, hogy annyi fejtorást okozzon, mint elődeiknek a nagygépeken...

A Griswold-féle makrónyelvre alapozott implementációk további előnye, hogy egészen kis terjedelműre szorítható össze a program tulajdonképpeni „magja”. A bonyolultabb dolgokat már le lehet írni magában a Snobol nyelvben, például külső függvények formájában. Ami megnézi a mágneslemezen közlő SNO.COM program méretét, csodálkozva állapíthatja meg, hogy feltűnően kicsi. Snobol nyelvben van írva már az interaktív használatot megkönykítő CODE.SNO program is, amely egyébként saját léteével egyrészt kézzelfogható bizonyítékát szolgáltatja a rendszermag bővíthetőségének, másrészt példát is mutat, hogyan lehet ezt megtenni.

Most akkor compiler vagy interpreter?

Logikádannak tűnik a „fordító-értelmező” összetétel? Általános fogalmaink szerint vagy az, vagy az, hiszen ez a két fogalom elég jelentős mértékben elterjed egymástól. A fordító futatható kódot készít, amely csak később kel életre, futtatáskor, ha különböző bemenő adatokkal „tápláljuk”. Az értelmezőről közismert, hogy nem különíti el egymástól a kódkészítést és a futás folyamatát, sőt a bemenő adatok és a környezet ismerete is annyira hozzátartozik magához az értelmezéshez, hogy nélkülük az el sem kezdődhet.

Hogy sarkosan fogalmazunk: az értelmező lassú és buta, hiszen mindig mindent újra kezd. Nemcsak hogy valamennyi futás alkalmával, hanem akár még egyetlen futáson belül is akárhányszor előfordulhat, hogy minden apró tennivalót (még a később föllesegésnek bizonyultakat is) unostanulantól újra fel kell ismernie, és szolgáiban végre kell hajtania. Az ilyen rendszer szervi hibája, hogy nem lát tovább az orránál — mondhatjuk valaki nem is alaptalanul. Vagy mégsem?

Nem lehet elvitatni, hogy a maga helyén az értelmező körülkenyessége nem is megvan a létjogosultsága. És nem is olyan biztos, hogy mindig igazságos ez a merev szétválasztás. Nem indokolt

már a LISP-nél sem, és még sokkal kevésbé a Snobol esetében.

Jók-e a skatulyák?

A LISP nyelv implementációjánál sem egészen a hagyományos értelemben beszélnek fordítóprogramról és értelmezőről. Nézzük előbb az értelmezőt. A LISP-implementációk „közönségesebb” változata maga is egy ciklikus üzemmódú interpreter, amely párbeszédet folytat a programozóval. Először türelmesen várja a feldolgozandó kifejezés bevitelét, majd mikor a „kész” jelet megkapja, a kifejezést kiértékeli. A harmadik fázisban kiírja a kiértékelés eredményét, azután újra kezd a türelmes várakozást.

Ami a fordítóprogramot illeti, az a LISP értelmezője számára nem más, mint egy beépített függvény, amelyet alkalmazni lehet valamilyen nem beépített függvényre. Az alkalmazás következtében — tulajdonképpen a kiértékelés mellékhatásaként — jön létre az a bizonyos gépi kód, amelyre várunk: az argumentumként megadott függvénynek a számítógép nyelvére lefordított változata. Alkalmazáskor ez az előre elkészített, gépi kódra lefordított függvény láthatatlanul helyettesíti azt a kifejezést, amely egyébként újra és újra értelmezésre szorulna. A fordítás és az értelmezés itt nincs szembesítve: amaz kiegészíti és „rövidre zárja” ezt. A végrehajtás így jelentősen felgyorsítható, éppen azért, hogy a lefordított függvények, eljárások szervesen beépíthetők az interpretálás folyamatába — ez szélsőséges esetben egyszerű futatással is korcsosulhat.

Ezáltal tehát jelentősen csökkenhetnek az interpreterek hátrányai, megmaradnak viszont az előnyei: az egyszerű programfejlesztés, az azonnali kipróbálhatóság, a módosítások könnyű megvalósítása.

Bizonyos szempontból analóg a helyzet a Snobollal is, hiszen interpreterének is van olyan beépített függvén-

nye, amely hasonló szerepet tölt be, mint a COMPILER függvény a LISP-ben. Itt azonban olyasmiről is szó van, ami miatt a Snobol „interpreterét” funkcionálisan meg kell különböztetnünk a hagyományos interpreterektől. Pontosan ezt kívántuk érzékeltetni, amikor a „fordító-értelmező” elnevezést használtuk.

A Snobol interpreterének tevékenysége szétválhat két szakaszra: „szintaktikázásra” és „szemantikázásra”. Az első szakaszban ez az okos kis jószág végigmegy a Snobol nyelvben írt programon, de a program előírásainak valódiság végrehajtása nélkül. Csak úgy mellesken viszont elvégzi a program szintaktikus elemzését. Sőt fordít is: lefordítja a programot egy gépfüggetlen belső kódra. Ez nyilvánvalóan azzal jár együtt, hogy közben feltárja a programot alkotó elemek egymáshoz kapcsolódásának belső szerkezetét is. És természetesen elküldi a felhasználónak a megfelelő hibajelzéseket. Ehhez megjeleníti a képernyőn a forrásprogram hibás részét, sőt lokalizálja a hiba jelentkezésének a helyét, és szövegesen kiírja a hiba típusát is.

A második szakaszra fatális hiba esetén már nem is kerül sor mindaddig, míg a programozó nem javít. Kisebb hibák esetén a rendszer azonnal megpróbálkozik a futtatással, hogy ezáltal több információ álljon rendelkezésre a hiba kijavításához. Azt talán már felesleges is hozzátenni, hogy a „szemantikázás” nyomán feltárt hiba jelegéről és előfordulási helyéről is tájékoztatást kap a programozó.

A kétszakaszos megoldás jelentős mértékben növeli a Snobol rendszer hatékonyságát: az első szakasz végtermékeként már kész egyrészt egy szintaktikailag hibamentes forrásprogramrészlet, másrészt ennek egy olyan belső kódú ábrázolása, amely már sokkal könnyebben és gyorsabban végrehajtható. (Talán nem is olyan biztos, hogy minden interpreter buta...)

Vargha Dénes

Megírja, hogy mit tart helyesnek?

Akár életfogytiglanit is kaphat!

(Cikkünk a 14. oldalon.)

MAXELL FLOPPYLEMEZEK NYÁRI VÁSÁRA!!

	1-10 doboz	11-50 doboz	51 doboz felett
	(Árak: Ft/doboz)		
MAXELL 5.25" DS/DD	700	670	640
MAXELL 5.25" DS/HD	990	960	920
MAXELL 3.5" HD	1800	1700	1650

Földelhető polarizált üvegszűrők
bevezető áron!

1-10 db	11-50 db	51 db felett
2250 Ft/db	1750 Ft/db	1450 Ft/db

ÁRAINK A 25% ÁFA-T MÁR TARTALMAZZÁK!



UNITRADE
Szervezési, kereskedelmi
és Számítástechnikai
K.F.T.

1073 Budapest VII., Erzsébet krt. 48.
Telefon/Fax: 142-2115

...nemcsak számítástechnika



TONER KFT

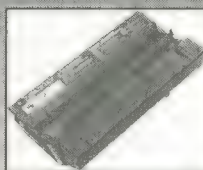


Másológép és
lézerprinter-kazetták
felújítása német technológia
alapján, garanciával,
lehetőleg és
színesben.



NE DOBJA EL!

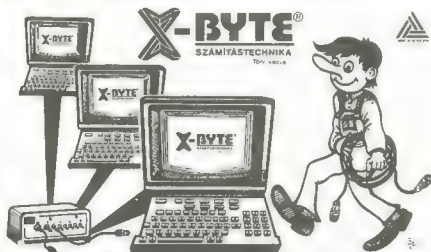
Felújítható
kazettatípusok:
FC, PC, EP,
EP-S, EP-L,
SHARP Z-30,
Z-50.



1095 Budapest Mester u. 21. Tel.: 113-1687 Tel./Fax: 134-3516

Europa International

INFORMÁCIÓKÉRÉS: 33



KAPCSOLÓDJON A JÖVŐHÖZ!

SZÁMÍTÓGÉPHÁLÓZATOK

MILYEN TÍPUSÚ HÁLÓZAT SZÜKSÉGES ÖNNEK?

ARCNET, ETHERNET, RS 232,
IBM CABLING SYSTEM AT & T,
SYSTEMAX, ÜVEGSZÁL,
NOVELL?

JÖJJÖN EL HOZZÁNK!

1138 Budapest, Népfürdő u. 17/e. Telefon: 173-1329 Fax: 173-1530

Egy kávé és üdítő mellett
segítünk a választásban.

CSÖKKENTETT ÁRAK. VÁLTOZATLAN MINŐSÉGI



Hun Comp

AKCIÓ!! AKCIÓ!!

- DATAS SVGA
(1024X768) monitor 25 000,- Ft + ÁFA
- AT 386/40 MHz számítógép
1.2 MB FDD-vel baby házban:
(monitor és wincsi nélkül,
egyébként kompletten) 49 500,- Ft + ÁFA
- 40 MB-os wincsi 19 000,- Ft + ÁFA
- 120-as wincsi 37 000,- Ft + ÁFA
- NOVELL 3.11 20 user: 199 000,- Ft + ÁFA

AKCIÓ!! AKCIÓ!!

HunComp

1116 Bp. Mohai út 37. Tel & FAX: 185-4186

INFORMÁCIÓKÉRÉS: 34

Számkígyó-bűvölés

Januári agytorna-gyakorlatunk, akár a decemberi, komoly kihívásnak ígérkezett.

Egyetlen kifejezés, egyetlen szám vagy számjegy előzetes ismerete nélkül kellett a megadott aritmetikai összefüggésekből meghatározni egy ismeretlen nyelv számrendszerének összes csínját-bínját.

Ráadásul még egy ijesztgetésnek is beillő figyelmeztetés vezette be a feladványt: hogy nem mindig előny, ha engedünk gondolkodásunk korábbi beidegződéseinek, és képtelenek vagyunk szabadulni „egyedül lehetségesnek képzelt gondolati formáinktól”.

Nosza, több sem kellett a gondolati formáktól szabadulni vágyó, szárnyalásra kész programozói fantáziának. Jó néhányan nemcsak az alapfeladatot és az extrafeladatot oldották meg elismerésre méltó találékonysággal, hanem még „szorgalmi” is beküldtek. Több „oda-vissza” programokat készítettek, amelyek az ún. arab számjegyekkel felírt számokat lefordítják az ismeretlen nyelvre (most már elárulhatjuk: a kínaire), illetve megfordítva: a sárkánykígyó hosszúsági kínai számnévekből elővárszólják „közönséges”, számjegyekkel felírt alakját.

A kínaik saját számnéveikkel írják a számokat — persze a számnévekhez tartozó hieroglifáikkal jelölve. Ma már terjed ugyan (főleg Tajvanon) az „arab” számjegyírás, de a hagyományaikra büszke kínaik még a lapszámozásban, és az évszámok jelölésében is máig ragaszkodnak szokásaikhoz.

Az osztás mint egyetemi tananyag

Ha már itt tartunk, szenteljünk egy pár sort a voltaképpen csak nevében arab számjegyírásunk történetének. Ami „arab”, az egyrészt eredetét tekintve indiai, másrészt az a mai arabok nem is ezt, hanem ennek egy igen-igen távoli rokonságot mutató változatát használják.

Vizsont Európába arab közvetítéssel jutott el az indiai számírás.

Először Abu Abdallah Muhammed ben Musza al-Hvarizmi (aha! az algoritmus akaratlan névadója) írt könyvet az indiai számírási módokról és az ehhez kötődő művelési szabályokról. Latin fordítása („De Numero Indorum”) tette

lehetővé, hogy lassacskán Európában is megismerjék a keleti számolóművészet titkait.

A bagdadi könyvtár rejtett kincseit feltáró matematikus, a horezmi születésű Mohamed érdeme mulhatatlan a numerikus számolás fejlődésében: a nyugati világban az ő munkásságának eredményeként terjedt el a helyi értékes, nullát is jelölő tízes számrendszer. Egyébként a „horezmi” maga még alig-alig használt szimbólumokat, a számokat is inkább kiírta betűkkel. A IX-X. században mégis a könyvből ismerték meg és kezdték tanítani az „új módot” a spanyolországi arab egyetemeken. A szabályokkal jól megfoghatóvá tett osztásnak például a mór kereskedők is hasznát vették, és a formalizálás aligha sikerülhetett volna a római számok alapján. A római számok technikájával 5000-tól leírni sem tudjuk a számokat!

Spanyolországi egyetemeken ismerte meg az osztás tudományát a francia Gerbert szerzetes is, aki később II. Szilveszter pápáként vonult be a történelembe. (Aha! Szent István koronája.) Meg is vádolták: nyilván az ördöggel cimborál, hiszen hogy lenne képes másképp bármekkora számokat elosztani egymással... Pápáként sem tudta eloszlatni a hagyományos jelölés híveinek ellenkezését. Hosszú, évszázadokig elhúzódó harc kezdődött a szabadelvű „algoritmikusok” és a konzervatív „abakuszok” között, és a XIII-XIV. századig inkább az utóbbiak voltak fölényben.

A fordulatot egy pisai fiatalember, Leonardo Pisano készítette elő a XIII. század elején. Műve, a „Liber Abaci”, vagyis az Abakuszok Könyve nem az

abakuszok használatát, hanem éppen ellenkezőleg, szükségtelességüket magyarázta el nagyon világosan. Leonardo, a Bonaccio-fiú — vagyis Fibonacci (van-e számítástechnikus, ki e nevet nem ismeri?) — számos gyakorlati példán mutatta meg, mennyivel könnyebb helyi értékes számokkal számolni, mint a hagyományos módon.

Számok — verselve és énekelve

A igen nagy és hosszú számok kiszámításában és megtanulásában is az indusok jártak az élen. Csillagászati könyveikben egész szinusztáblázatokat foglaltak versebe. Írásos hagyományaik szerint a tíz 140-edik hatványára is volt szavuk (asamkhyeya). Széles körben használt rendszerről azonban nem beszélhetünk, hiszen például a nagy számok megnevezését illetően nem egységesek a különböző források. Az iskolák mindegyike a maga hagyományait ápolta. Tény viszont, hogy a számok potenciális végtelensége és a nagy számok közötti rendcsinálás régóta izgatta az indiai matematikusokat. Bizonyára ez vezetett ahhoz, hogy voltaképpen rájöttek a rekurzív definíció lehetőségére.

A ma élő nyelvek közül a kínaiban található „természetes nyelvi eszköz”, tehát külön tőszó a legnagyobb természetes szám kifejezésére. Ez a legnagyobb szám a százmillió, vagyis a tíz 8-adik hatványa. A nagy számok tagolása elterjedt a miénktől: nem hármasával (illetve hatosával) tagolják a számjegyeket a kínaik, hanem négyesével. A százmillió is azért lehet tőszó, mert tízezer-szer tízezer.

Ezt a számot írjuk feladatunkban Yü-nek, hogy világosan megkülönböztessük az '1' jelentésű Yi-től. Valójában ez is Yi, csak „másképp éneklük”. Zeneileg egyvonalas c-vel jelölhetjük az '1' jelentésű Yi-t, amelynek tónusa kitarított (1. tónus), a '100.000.000' jelentésű Yi viszont ereszkedő tónusú (4. tónus), amely ugyanerről a hangról indul, de kis szétlépéssel leereszkedik a kis e-re (szolmizálva: mi-do).

Kegyetlenül nehéz hangtani átírásban visszaadni a kínai nyelv kiejtését. Minden magánhangzónak négyféle tónusa lehet: kitarított, emelkedő, ereszkedő-emelkedő és ereszkedő; a tónusútl függően más-más lehet a szavak jelentése. A „ma” szó négyféle jelentése: 1. anya, 2. kender, 3. ló, 4. káromkodni. (Micsoda táptalaja a félreértésnek!)

De ez még mind semmi. Mert ha a tónusokat pontosan megkülönböztetjük is, akkor is marad rengeteg azonos

hangzású szó, amelyek kiejtésükben nem térnek el egymástól, csupán a hieroglifájuk különbözik. Például a Yi szónak az első és a második tónussal 7-7 különböző hieroglifa felel meg, a harmadikkal 5, a negyedikkel — nem tévedés — 21! Nem csupán hagyománytszisztelet vezeti hát a kínaikat, amikor nem egykönnyen mondanak le megszokott frászmódjukról.

Kínai számok kódolása és dekódolása

„A bevezető alapján a legvadabb dolgokra gondoltam: nem tízes számrendszer, római számokhoz hasonló kivonásos számképzés...” — írja Csurgay Péter Cseperlől.

Legprecízebb megfeytőnk, Dezső András is fel volt készülve minden rosszra: „Szinte biztos voltam benne, hogy még csak nem is tizenkettes számrendszerrel van szó. Képtelen voltam szabadulni saját, egyedül lehetségesnek képzelt gondolati formáimtól!” — emlékezik.

Süle Gábor is a rendszer alapszerkezetében várt különbb gondolatok kényszerét: „Arra számítottam, hogy valami nagyon egzotikus vagy legalábbis bizarr számrendszer áll az idegen szavak mögött. Utólag azt gondolom: jóval többet lehetett volna bajlódni mondjuk a római számok vagy a francia számnevek áruhába bújtatott képviselőivel. Ezer (akarmondani: tízezer) szerencse, hogy nem róluk volt szó.”

Mégsem volt hiábavaló az ijesztgetés. Még így is előfordult, hogy valaki jól „dekódolta” ugyan a feladatban rejlő üzenetet, amikor azonban „kódolni” kellett volna, vagyis a felismert szabályszerűségeket megfelelő „inverz transzformációkat” kellett volna alkalmazni, az már nem sikerült. Legtöbbször éppen a négyjegyű tagolás szokatlansága miatt.

A megfejtés

A legnagyobb körültekintéssel Dezső András oldotta meg a feladatot, az ő gondolatmenete alapján ismertetjük a megfejtést (programját mágneselemző mellékletünkön is közöljük).

„Az első információcsokor tanulmányozása után egy dologban lehettem csak biztos: mást jelent a SiShi, mást a ShiSi. Ebben a hitemben megerősített a harmadik kérdés is. A második csoportból először (6)-ra lettem figyelmes. Tudván, hogy Er nem nagyobb 10-nél, Yi-nek egynek vagy kettőnek kell lennie, különben a bal oldal nagyobb lenne tiznél, ez pedig ellentmondana a fenti

állításnak. Tehát Yi = 1 és Er = 2, vagy Yi = 2 és Er = 6. Ezekkel az információkkal azonban tovább nem jutottam.

További megfigyeléseim során három csoportra osztottam a szócskákat. Az egyik fajta az, amelyik nem szerepel összetételekben. Ilyen a Yü, a Wan, a Qi és a Ling. Később kiderült, hogy a Qi másik csoportba tartozik, sőt a Ling is kakukkfíók. A másik fajta, amelyik szinte mindig csak összetételben található. Ezek a Qian, a Bai és a Shi. Őket szinte mindig megelőzi egy harmadik csoportba tartozó szó is (Yi, Er, San, Si, Wu, Liu, Qi, Ba, Jiu). A Shi külön: előtte is, utána is elhelyezkedhet a szócska, sőt állhat néha egyedül is. A Bai csak a Wan előtt magányos.

(10)-ból és (5)-ből felismertem, hogy Wan és Yü négyzetszámok, sőt mivel Yü a Wan-nak a négyzete, Yü negyedik hatvány. Lehet 16, 81..., de valószínűbb, hogy 10 000. Ennél nagyobb számra gondolni nem mertem. Reménykedni kezdtem, hátha mégis tízes alapú rendszerdek az ismeretlen nyelv számnevei.

(12)-ből kifejeztem Bai_Wan-t, (9)-ből Yü-t:

Bai_Wan = JiuShi_Jiu_Wan + Wan
Yü = JiuQian_JiuBai_Wan + Bai_Wan,
majd a második egyenletben behelyettesítettem Bai_Wan-t:
Yü = JiuQian_JiuBai_Wan +
+ JiuShi_Jiu_Wan + Wan.

Látászik, hogy ha a sok Jiu...-hoz hozzáadunk egy Wan-t, a Jiu-k sorra többszörösek. A Jiu tehát a legnagyobb számjegy. Qian, Bai és Shi „kis” helyi értékek, Yü és Wan pedig „nagy” helyi értékek. Yü a Wan-nak a többszöröse, négyzete. Közöttük négy helyi érték különbség van. Ez azzal magyarázható, hogy Wan tízezer, Yü pedig százmillió. Qian így ezres, Bai százás, Shi tízes helyi értéket jelöl.

Feltevéseimet (15) is igazolni látszott. A számjegyek tehát nem hármas csoportokra rendeződnek, mint a magyarban, hanem négyesekbe. Ha a csoport után Yü vagy Wan áll, úgy a csoport számértékét szorozni kell százmillióval, illetve tízezerrel. Ha nincs utána semmi, akkor ez már a csoport számértéke. A csoporton belül a helyiértékszó határozza meg az adott helyi értéken lévő számjegyet. Ha a csoport középső helyi értéke közül valamelyik (vagy mindkettő) nulla, akkor egy Ling iktatódik be a nem nullák közé.

A számjegyeket már könnyebb volt megtalálni ezek után. (13) bal oldalán 90 + 90 áll, jobb oldala tehát 180. A tízes helyi értéken álló Ba tehát a nyolc, a százason álló Yi az egy. (6) miatt

ekkor Er = 2. (11) jobb oldala 22, ezek szerint ShiYi tízenegyet jelent. A tízenvalahány úgy képződik, hogy a Shi után kerül a számjegy-szó. (4) bal oldali értéke 16, innen Liu = 6. (7)-ben egy szám négyzete meg egy egyenlő egy kerek számmal. Mivel a Shi nem önmagában áll, tehát a bal oldal nem tíz, ezért Qi nem lehet három, csak hét. (3) bal oldala 2 * 7 = 14, ebből Si = 4. (1)-ben valahányszor kettő egyenlő tízzel, Wu = 5. Már csak a három hiányzik: 19 bal oldalán 11 harmadik hatványa egyenlő 1331-gyel, San = 3.

Körülbelül ily módon tárulkoztott elém az igazság. Ezek után a biztonság kedvéért ellenőriztem minden egyenletet: működött...”

D. A. gondolatmenetének empirikus igazolását jelentik az általa készített konvertáló programok. Van, aki rövidebb programmal oldja meg a konvertálást (például Csurgay Péter — ő nagyon ügyesen használta ki a rekurzióban rejlő lehetőségeket), D. A. programja azonban pontosabban kezel minden lehetőséget, és módszeres ellenőrzéssel kiszűri a felmerülhető hibákat. Igen alapos konvertáló programokat készített Süle Gábor is FoxPro 2.0-ben. Csaszny Márton részletes blokkdiagrammal írta le a megoldást, Domszky Zoltán pedig 14 lépésben foglalta össze a program működésének különböző állapotait: mikor mit kell vizsgálni, mit és milyen sorrendben kell megvizsgálni, mit kell felírni, mi a foglalt tárolókerek tartalma. Mindegyiküknek gratulálunk, és sajnáljuk, hogy nem lehet tapasztalatszerként valamennyiük megoldását közölni.

A megfejtéseket azonban nem mondjuk meg — tessék megkérdezni a programról.

Vargha Dénes

Lapzárta után

A Novell cég szeptemberben ismét hasonló előadássorozatot tart, mint a júliusi Network University „Road Show” volt, s amelynek tapasztalatairól lapzártánk miatt csak a szeptemberi Alaplapban tudunk beszámolni. A téma ezúttal a hálózatos adatbázis-kezelők alkalmazása lesz. A Novell bevonja az oktatásba a Gupta Technologies és az Oracle szakembereit is.

Valami azért marad belőle

Volt egyszer egy PC Turbo Klub

Bizonyára többeknek feltűnt, hogy a PC Turbo Klubba invitáló válasz-levelező-lap szokásos helyét mostani számunkban valami más foglalta el. Ez a „valami más” eddigi koncepcióknak módosítását jelenti. A PC Turbo Klub funkcióját, működési formáját átalakítottuk.

Az 1991 elején indított speciális előfizetési forma olvasóink körében gyorsan népszerűvé vált. A számonkénti 20 forintos engedmény és az áruházi vásárlási kedvezmény mellett sokan fogadták örömmel a szakmai bemutatókat, a konzultációs lehetőségeket, az információszolgáltatást. Végére is nem rossz dolog olyan klub tagjának lenni, amely csak szolgáltatásokat és kedvezményeket nyújt, a klubtagokat pedig semmilyen kötelezettség nem terheli.

Anélkül, hogy finánciális részletekkel terhelnénk olvasóinkat, el kell mondanunk azt, ami mostanra számunkra egyértelművé vált: a klub bevételi forrásai mára már nem fedezik azokat a többletárfordításokat, amelyeket a rendezvények szervezése, a nyilvántartás adminisztrációja, az expedálás, a reklamációk intézése jelent, nem is szólva az emelkedő postai tarifákról.

A mi esetünk tanulságos lehet a hasonló cipőben járó többi szerkesztőség számára. A lap fennmaradása érdekében minimális létszámmal dolgozó szerkesztőségi és kiadói gárda ne törekedjen olyan terjesztési funkciók átvállalására, amelyek hatékonyabban elvégeztethetők a terjesztésre szakosodott nagyobb szervezetekkel. Mi legalábbis erőnkön felüli feladatra vállalkozunk a klub működésével, és a rendezvények szervezésére vagy külön szolgáltatások nyújtására nem jutott elegendő energiánk, a lap készítése túlságosan leterheli a 4 fős főállású szerkesztőségi gárdánkat.

Aki eddig jutott az olvasásban, nyilván úgy gondolja: „Na, kész, ezek is becsukják a kaput, nincs tovább klub, nincsenek kedvezmények, vegyem meg az A-lapot az újságárusnál vagy fizessen elő a Postánál.” Ami az előfizetést illeti, valóban nincs más lehetőség, mint visszatérni a Postához. Ez a váltás anyagiilag számunkra is bizonyos veszteséget okoz, mégis vállalnunk kell, hogy minden erőnket a lap jobbfi-

tására fordíthassuk. Az igazsághoz tartozik még, hogy az előfizetők többsége a PC Turbo Klub létrehozása után is kitartott a hagyományos postai előfizetési mód mellett. (Szerencsére!)

Azt is be kell vallanunk azonban, hogy egy kicsit fáj a szívünk ezért a PC Turbo Klubért. Valami hasznos és jó dolgot kezdeményeztünk, ami által személy szerint is közelebb kerülünk lelkes és hűséges olvasóinkhoz. Sokáig törtük a fejünket, hogyan lehetne megmenteni valamit ebből a klubból. Arra a következtetésre jutottunk, hogy nagyobb munkaráfordítás nélkül megőrizhetjük — sőt kiterjeszthetjük — szakmai közvetítő szerepét, ha a tagok címét adatbázisba szervezzük. Úgy véljük, hogy a címanyagot felhasználó cégek-től olvasóink olyan információkat is megkaphatnak,

amelyek közlésére a lapban nincs lehetőség. (Például helyi és regionális rendezvényekre szóló meghívók.)

Ez a lista akár a teljes olvasótáborra felölthető. Úgy képzeltük el, hogy mindenki eldönti, akar-e szerepelni e megváltozott funkciójú klub nyilvántartásában. Aki igen, az mindössze annyit vállal, hogy címét megadjuk az értékesítési, konferenciaszervezési vagy más szakmai céllal címanyagot kereső cégeknek, szervezeteknek. A Cédus Karolina Áruház és a Floppyland Kft az adatbázisban szereplőknek pedig továbbra is 5%-os vásárlási kedvez-

ményt ad. Az eddigi PC Turbo Klubtagok automatikusan bekerülnek a címadatbázisba, hacsak a kartonból kivágható válasz-levelezőlapon vagy más módon nem kéri ennek ellenkezőjét. Aki nem volt tagja a PC Turbo Klubnak, az viszont kérheti, hogy neve és címe szerepeljen ebben az új listában. A választadásra nem szabunk határidőt, mert a klub teljesen nyitott, bárki bármikor kérheti a címanyagba való felvételét vagy onnan való törlését.

Végül pedig alapvető szolgáltatásunkat illetően a „régii” klubtagok mindaddig továbbra is közvetlenül tőlünk kapják az A-lapot, amíg „le nem cseng” az előfizetésük. Az előfizetések meghosszabbítása azonban már csak a Postánál lehetséges.

Varga János

CHICONY AT-486DX/33 LAPTOP

- 2 Mb RAM
- 1,44 Mb floppy
- 60 Mb winchester
- Gázplazma VGA display (16 árnyalat)
- S/P kimenet
- külső 5,25" floppy csatlakozó
- külső VGA monitor csatlakozó
- súlya 4,8 kg
- 2 órás akkumulátor üzemidő
- 1 db 16 bites szabad kártyahely

299.000,-Ft

SVGA MONITOR

(felbontás:1024x768, 0,28 dp, HITACHI képcső)

29.000,-Ft

TSENG-LAB VGA KÁRTYA

(ET 4000 processzor, 1Mb RAM)

9.000,-Ft

1 év garancia!

Áraink az ÁFA-t nem tartalmazzák!



DATA DOCTOR Kft.

1149 Bp. Buzogány utca 4. Tel./Fax : 183-72-99

Mikrogyépgyártás

Amerikai számítógépgyártók egy csoportja a mikro-számítógépek előállításai adatainak nyilvántartására alakított bizottságot. Japánban már működik hasonló szervezet, amely az utóbbi tíz évre gyűjtött iparági adatokat, Nyugat-Európában pedig most szerveznek egy ilyen csoportot. Az amerikai tömörülés munkájában többek közt részt vesz az IBM, az Apple és a Compaq. A bizottság negyed-évente kívánja ellátni tagjait a mikroszámítógépek gyártására vonatkozó adatokkal, válaszolva a fontosabb trendeket.

Új könyvtár a BME-n

Mintegy negyvenmillió forint felhasználásával korszerűsítették a Budapesti Műszaki Egyetem könyvtárát. Ez nem elsősorban az új olvasóter kialakítását jelenti, hanem az integrált számítógépes rendszer bevezetését. Lelkét az izraeli El Libris cég által kifejlesztett és szállított Aleph nevű szoftver jelenti. Ez kifejezetten egyetemi környezetben készült, hasonló működik Európa negyvenöt helyén, többek között a dán műszaki egyetemen, a genfi nukleáris kutatóközpontban, valamint New Jersey Princeton egyetemén. Izraelben minden egyetem ezt a rendszert használja. Hasonló célt tűznek ki Magyarországon is.

Judy

Elkészült a Judy szoftvertervező és programíró rendszer új verziója, mely már Turbo Pascal 6.0

forráskódú programot is ír. Alkalmazásával a felhasználói programok széles körét elegendő csupán — interaktív üzemmódban — vázlatosan megtervezni, s a rendszer másodpercek alatt megírja a forrányelvű programot. Az így kapott eredmény jellemzői: a Turbo Pascal vázlatított verziójában szintaktikailag hibátlan, strukturált, futási időre, tárra optimalizált programok keletkeznek. A Judy által írt programok képesek más, forrányelvű fájlokból eljárásokat, függvényeket beszúrní, használni építő, mint lefordított (.EXE vagy .COM) programokat meghívni. A létrehozott programok más programnyelvekben — például Clipper, C, Cobol — is elnyövesen használhatók. A kész programok később akár-hányzsról módosíthatók. A rendszer megvizsgálja a hardvert, és a tulajdonságait a generált program automatikusan örökli, azaz öninstallálódóvá válnak. Így például automatikusan generálódnak az edgekezelési funkciók a leendő programba. A létrehozott programok .PAS kiterjesztésű fájlokba kerülnek, s éppen olyanok, mintha szövegszerkesztővel írtuk volna őket. A Judy már jelenleg is tantervi anyag a Kandó Kálmán Műszaki Főiskolán.

Számítástechnika farmereknek

A mezőgazdasági kisüzemek számítástechnikai hátterének megteremtését tűzte ki célul az Agrog Kft. A tapasztalatok szerint ugyanis napjainkban már a hazai mezőgazdasági közép- és kisüzemek sem nélkülözhetik az a biztonságot, amelyet külföldön a korszerű számítástechnika nyújt a termelőknek. E hátter hazai kialakításának megalapozásaként költöttek szerződést a Siemens Auszria céggel. Ennek értelmében megkezdik az osztrák cég által kifejlesztett, s a farmergazdaságokban jól használható programrendszer forgalmazását. Ez egy számítógéppel támogatott termelésirányítási és -elszámolási rendszert foglal magában, amely felhasználható többek között a növénytermesztés és az állattenyésztés irányítására, és alkalmas az ügyviteli munka, többek között a főkönyvi könyvelés elvégzésére.

zések pótlására szolgált. 1990 végén több mint ötezer üzemben 17 ezer számítógépes munkahely üzemelt, 6500-zal több, mint az előző évben. Svájcban már 1990-ben is 1000 munkavállalóra 180 mikrogép jutott. Ezzel Svájc megkezdte a világszerte a számítógép-sűrűség tekintetében.

Jóslat az ezredfordulóra

2000-re a fejlett tőkés országokban a foglalkoztatottak negyven százaléka számítógépes munkahelyen fog dolgozni, és minden ötödik munkavállalótól elvárják a főnökei, hogy kreatív módon kezelje a gépet, azaz az egyszerű szoftverhasználatnál többre is képes legyen. Ez a jóslat hangzott el a tekintélyes osztrák Der Standard című folyóiratban. A fejlődés azonban korántsem ennyire egyértelmű. A számítógépiparban ugyanis a fejlődés ütlősságban gyors, az emberek tömegei egyszerűen nincsenek felkészülve arra, hogy az egyre bonyolultabb információrendszerekben eligazodjanak, annak összes lehetséges következményeit és hatásait felmérjék. A fejlődés geográfiai motorjaként elismert japánok és amerikaiak mellett immár csaknem bizonyosra vehető, hogy az európaiakra csupán a néző szerepe marad.

Számítógép a bűnözők között

A BM és az SZKI Pixel együttműködésével kifejlesztett rendszer a számítógép mesterséges intelligenciáját hívja segítségül a tettesek felkutatásához. A lényege, hogy — a hagyományos mozaikkép-készítési eljárással szemben — a tanú által kiválasztott, a tettesre leginkább emlékeztető arcnszerűlet elképesztő valóságssággal illeszkedik a monitoron lévő archív. A program bármilyen képtranszformációs műveletet elvégez, nagyítási, kicsinyítési, átméretezési, rőncosítási, tetszés szerint. Az így kapott portré fényképszerű ábrázolást ad a tettesekről. Közel tíz rendőrkapitányságon használják már a képfeldolgozó rendszert, meghozható sikerrel: jó néhány veszedelmes bűnöző akadott horogra az ekképp készült portrék segítségével.

Hasonlóképpen a bűnözők segítőségére lehet a BM és a Geometria együttműködésével kifejlesztett térinformatikai rendszer is. Ez tematikai szempontok szerint képes megjeleníteni az adott városrészt épületeinek alaprajzait, csatorna- vagy elektromos hálózatát, megmutatva például a bővíthető lehetőségek menükülső útvonalait. Ha a számítógépbe beviszik, hogy hol, mikor, mifele bűncselekményeket követtek el a városban, akkor a rendszer — megkeresve az összefüggéseket — tájékoztatja a rendőrokat a város rablási szempontból legfertőzöttebb pontjairól. Ehhez igazodva azután a rendőrség hatékonyabban tervezheti meg például a járőrszolgálatok útvonalait.

Végezetül essék még szó egy, szintén a mesterséges intelligenciára épülő, ám ma még igen-csak a tervezési állapotban lévő ötletéről. Az ORFK a Számak Multilogic Kft.-vel együtt gondolkodva egy olyan, nyomonzást támogató, szakértői rendszer kidolgozásán fáradozik, amely segítséget nyújt az egyes bűncselekmények felderítéséhez. Számítógéppel feldolgozva például a nyomozóknak a helyszíni szemlélek kapcsolatos évtizedes tapasztalatait összevetve lehet éppen aktuális helyszíni szemléni összegyűjtött információkkal, a gép pedig lehetséges alternatívákat adhat a nyomozás elindításához.



COMPUTER

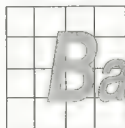
2 év garancia,
fél év alaplapp-upgrade,
alacsony sugárzású monitor.

MANNESMANN

nyomtatók

teljes választéka.

Nagy megbízhatóság.



1054 Budapest V.,
Bajcsy-Zsilinszky út 54.
Tel./Fax: 111-6025

1054 Budapest V.,
Bajcsy-Zsilinszky út 64.
Tel./Fax: 131-1960

Számítógép-sűrűségi világsúcs

Svájcban 1990-ben az állami hivatalokban és a gazdaságban alkalmazott számítógépek száma 130 ezerrel nőtt, s elérte az 530 ezret. Igazság szerint 156 ezer számítógépet helyeztek üzembe, de ezek közül 20 ezer a kiöregedett berende-

A MikrobaZár rovatban rövid, szöveges, a mikroszámítógépekkel kapcsolatos hírdetéseket közlünk.

A kereskedelmi tevékenységet szolgáló apróhirdetések tarifája gépielt soronként (60 karakterenként) 300 Ft. Kérjük, hogy a hirdetés díját a Cédrus Kiadó Kft-nek az Általános Értéktörzsi Banknál vezetett 204-19417 számú számlájára utalják át, vagy postautólevélvonalon fizessék be a Cédrus Kiadó Kft címére (1441 Budapest VIII., Reguly Antal u. 8.), a hátoldalon feltüntetve, hogy apróhirdetés. A befizetést igazoló szelvényt a közlendő hirdetés szövegével együtt az Alaplap szerkesztőségéhez (a kiadóval azonos címre) küldjék el.

A nem kereskedelmi célú egyéni hirdetések közlése INGYENES!

ELAD

Eladó Commodore 64-es gép magnóval, kazettákkal, joystickkal, lemezekkel. Ár: 13 000 Ft. PC-s programcsere is érdekel. Cserealap: Street 1-2, teljes Sim-sorozat, Flight Sim V3, stb... Keresem a Jones és a Lemmings elnevezésű programokat. Cím: Bozsó G. István, 7400 Kaposvár, Kisközi u. 18.

3 éves Commodore 64-es gép olcsón eladó. Ár: 9991 Ft. Cím: Jozsefák János, 1181 Budapest XVIII., Háromszék u. 15.

Eladók Commodore 64-hez az alábbiak: 1802 színes monitor, 1541 floppy, 1535 datencorend, Final Cartridge III., valamint 10 db Data Becker könyv. Ár együttesen 26 000 Ft. Cím: Kun József, 1213 Budapest XXI., Puli sétány 19. III./14. Tel.: 276-7225.

Eladó Sinclair Spectrum számítógép összes tartozékával, 128 kB memóriával. Cím: Dietrich András, 8700 Marcali, Széchenyi u. 16. IV./13.

Eladó IBM XT: 640 kB RAM, 20 MB winchester, 360 kB floppy, Hercules kártya, 14"-os monokrómon monitor, 101 gombos billentyűzet. Ár: 25 000 Ft. Cím: Oláh Andre, 1111 Budapest XI., Irinyi J. u. 9-11. 866-os szoba (este 18-20-ig).

Eladó AT 286: 1 MB RAM, 1,2 MB floppy, monokrómon monitor, 101 gombos billentyűzet. Cím: Lovas Tibor, 7030 Paks, Becsényi u. 46. Tel.: (75) 14-313.

Eladók programleírások Commodore, Amiga, Spectrum és IBM gépekhez. Válaszboríték ellenében leírást küldök a kiválasztott programról. Cím: Heszler Róbert, 8220 Batatonalmádi, Móra F. u. 5. II/8.

PC video vezérlőkártyák (Hercules, CGA, EGA, VGA) hardveres ékezetesítése CWI vagy egyéb tetszőleges kódkészlet szerint. PC Turbo Klubtagoknak kedvezmény! Cím: Lóth Tamás, 1558 Budapest, Pf. 76.

Számítástechnikai oktatás IBM PC gépen bármilyen témában a legolcsóbban! Beszerzési tanácsadást és programkészítést is vállalok! PC Turbo Klub-tagoknak kedvezmény! Cím: Fridl György, 1116 Budapest XI., Szalóki u. 6.

K-ÉP Stúdió építészeti CAD-szoftvert és kereskedelmi-ügynöki feladatokhoz. 5-20%-os jutalékkal Cím: K-ÉP Stúdió, 1426 Budapest, Pf. 66. Tel.: 133-4760/328-as mellék (Kuczogai László).

Objektumorientált programozás Clipper 5.01-ben: Objects 1.1. Kérésre tájékoztatót küldök. Cím: Szűcs János, 4400 Nyíregyháza, Vasvári Pál u. 37. IV./13. Tel.: (42) 13-568. Programokra, segítségre, cseretársakra van szükség? Az ASIS megoldja problémáit! Bárhol laksz, bármilyen géped van, írj! Kérésre ingyenes tájékoztatót küldök. Cím: ASIS, 1425 Budapest, Pf. 729. Tel.: 142-8075.

VESZ

Vennék jelképes áron használható állapotban levő ZX Spectrum gépet tartozékokkal és magyar nyelvű leírással. Cím: Helfrich Zoltán, 1062 Budapest VI., Bulcsu u. 8.

CSERÉL

Amíg a zenemodulokat, Roland MC-50-re songokat, MSX-re programokat cserélek vagy adok. Cím: Jancsúrák István, 3528 Miskolc, Dráva u. 7. Tel.: (46) 380-251.

PC-n friss stuffokat cserélünk, valamint alakuló csapatunkba tagokat keresünk: grafikus, zenész, swapper, assemblyben programozó munkatársak jelentkezését várjuk. Ugyanitt Adlib hangkártya 20 000 forintért eladó. Cím: Fábrián Zoltán, 8500 Pápa, Martinovics u. 9.

Cserélünk IBM XT/AT programokat. Válaszboríték ellenében listát küldök. Ugyanitt eladó ZX-81 tartozékokkal. Ár: 3000 Ft. Megrendelhető továbbá IBM AT-hoz soros hangadapter. Ár: 800 Ft. Cím: Grexa Miklós, 2628 Szob, József Attila út 3.

Bibliográfia

A hónap témájához az 1991/4. számunkban megjelent összeállításunkat ajánljuk olvasóink figyelmébe. Most egy 13 kötetesre tervezett sorozat eddig megjelent kötetének adatait ismertetjük. A sorozat kiadója a Miskolci Egyetem Dunajvárosi Főiskola Kár Kádó Hivatal.

Dr. Ferencik János: Az IBM számítógép használata 1. (A gép — Programok — A gép be- és kikapcsolása — DOS parancsok — Program beírása, indítása — A KEYDEF program — A PCTOOLS program) Dunajváros, 1992. 152 oldal. Ár: 200.— Ft.

Dr. Ferencik János: Az IBM számítógép használata 2. (Szövegszerkesztés, a CHWWriter programcsomag) Dunajváros, 1992. 100 oldal. Ár: 190.— Ft.

Dr. Ferencik János: Az IBM számítógép használata 3. (Táblázatkezelés, a Lotus 1.2.3. programcsomag) Dunajváros, 1992. 181 oldal. Ár: 200.— Ft.

Dr. Ferencik János: Az IBM számítógép használata 4. (Tervezés, szerkesztés az AutoCAD programcsomag) Dunajváros, 1992. 276 oldal. Ár: 375.— Ft.

Dr. Ferencik János: Az IBM számítógép használata 5. (Profi billentyűzetindítás: KEYBPPAT — Adattárolmányok kezelése: dBASE III, PLUS, CLIPPER) Dunajváros, 1992. 232 oldal. Ár: 290.— Ft.

Dr. Ferencik János: Az IBM számítógép használata 6. (Nyomtatás az EPSON FX 1050 nyomtató — A Norton program: Commander, Integrator, Editor, Guide — A SIDEKICK titkár(nő) program — az IBM PC gépek helyi hálózatban: a NOVELL hálózat — Vírusok, víruskezelés. Számok, vírusleírás) Dunajváros, 1992. 207 oldal. Ár: 255.— Ft.

Dr. Ferencik János: Az IBM számítógép használata 7. (Programozás BASIC nyelven: a GWBASIC program) Dunajváros, 1992. 200 oldal. Ár: 265.— Ft.

Dr. Ferencik János: Az IBM számítógép használata 8. (Programozás PASCAL nyelven: a TurboPASCAL program) Dunajváros, 1992. 338 oldal. Ár: 430.— Ft.

Dr. Ferencik János: Az IBM számítógép használata 9. (Kiadványsszerkesztés: a Ventura HVP program — Képek, rajzok, szövegek, programok beírása: ecmrnr) Dunajváros, 1992. 304 oldal. Ár: 405.— Ft.



HOKTRADE Co. Ltd.
INDUSTRIAL AND COMMERCIAL CO. LTD.

1012 Budapest I., Attila út 93.
Tel.: 202-4166 • Fax: 175-0446

PC-alkatrészek nagy választékban

286-os, 386-os, 486-os konfigurációk alacsony áron

SS 212 printermegosztó	4 500,—
Babypárház + táp, LED kijelzővel	4 990,—
Toronyház + táp, LED kijelzővel	9 990,—
CITIZEN MSP 40 printer	15 900,—
Külső modem 2400M5	12 500,—
Modemkártya 2400M5	11 000,—
Pocket modem 2400M5	13 000,—

A fenti árak a 25%-os áfát nem tartalmazzák.



INTRAM Szerviz és Kereskedelmi Kft.

1072 Budapest VII., Kis Diófa u. 6.
Telefon: 122-0087 Fax: 121-3230



Ilyen még nem volt Magyarországon!

Everex és Wyse számítógépek a profiknak, akiknek csak a legjobb elég jó
Olcsó amerikai számítógépek azoknak, akiknek számít, mit kapnak a pénzükért
Sysdoki és Sysguard mindenkinek, akinek fontos a vírus- és adatvédelem

Érték ügyviteli- és vezetői információs rendszer
azoknak, akik tudni akarják, hogyan áll a cégük

Apróclikkek, meglepetések, kedvezmények mindenkinek, aki szeret jót vásárolni

Nálunk a minőség mindig megéri az árat!

Intelligencia minden szinten

E havi összeállításunk egymástól igencsak távol eső termék- és szolgáltatáskínálatot vonultat fel — a hálózatépítési módszerektől kezdve a COCOM-rendelkezők könnyítésén át az adatvédelemmel és a mesterséges intelligenciával kapcsolatos hazai kutatás kimagasló eredményeinek bemutatásáig.

A közös vonás, amely alapján rokonítható e négy terület: a potenciális felhasználó igényelmének szolgálata.

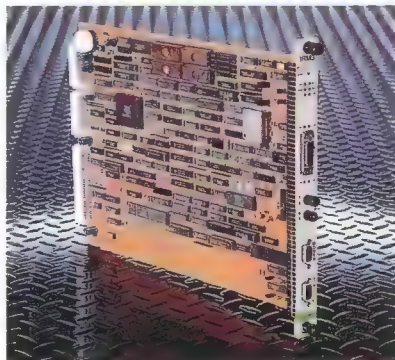
Válás után újabb frigy

Alig egy éve számoltunk be arról, hogy a hálózati termékek forgalmazása területén a Rolitron megállapodást kötött a BICC-vel. A házasság azonban rövid életűnek bizonyult. Ugyanis a pénzügyi nehézségekkel küszködő — de kiváló technikai képességekkel rendelkező — BICC-t felvásároló 3Com filiozódja alaposan eltér a rendszerintegrátor Rolitron koncepciójától.

A Rolitron újonnan választott partnere erős technikai háttérrel és komplex termékalkálával rendelkezik. A Cabletron a világon a harmadik helyen áll (a Synoptics és az Ungerman Brass után) a hálózati elemek forgalmazásában. A Cabletron-termékek a hálózatokkal kapcsolatos valamennyi igényt kielégítik. A Cabletron tetszőleges számú számítógép között létesít hálózati kapcsolatot, legyenek azok a világ bármely részén. Egyaránt vállalkoznak Ethernet, Token Ring és FDDI lokális és nagy távolságú hálózatok kiépítésére. Választékukban teljes a hálózati kapcsolóelemek és szerelési anyagok repertoárja: kábelek, transceiverek, menedzselhető hálózati csatlakozók, intelligens moduláris felépítésű hálózati elemek (hubok, repeaterok, lokális és remote bridge-ek) mellett az alkatrészek, kiegészítők és tesztműszerek alkotják a Cabletron termékpalettáját.

Tekintettel arra, hogy a Cabletron mindent maga gyárt — megbízható minőségben és alacsony áron —, az értékesítéskor nincsenek beszállítási problémák, és rövidek a szállítási határidők.

A Cabletron-termékek szabványosak. Kezdetben csak követték a szabványokat, ma már viszont ők írják azokat. A velük kapcsolatban álló cégek (Novell, Banyan...) úgy alakítják termékeiket, hogy azok illeszkedjenek a Cabletron-palettára. A gyártás az Integrált Hálózati Architektúrán alapul (INA). Nagy segítséget jelent a felhasználóknak, hogy a Cabletron-termékek nemcsak egymással, hanem más cégek termékeivel is kompatibilisak. Így globális, osztott hálózatok építhetők ki és irányíthatók olyan termékekkel, amelyeket hálózati szabványtól függetlenül egy közös rendszerben működtetnek együtt.



A Cabletron háromféle hálózatiirányítási eszközt is kínál (MMAC, Remote Lanview, Spectrum). A hardveralapú, moduláris Multi Media Access Center (MMAC) az Ethernet és FDDI (űvegsvál) hálózati elemek mellett Token Ring szegmenseket is képes hálózatba egyesíteni. A Remote Lanview olyan hálózatiirányítási szoftver, amely figyel és ellenőrzi a hálózati kapcsolatokat, valamint diagnosztizálja a felmerülő problémákat. A Unix-alapú Spectrum pedig a különféle hálózati topológiák támogatása mellett figyel és ellenőrzi a különböző gyártók termékeiből kialakított heterogén hálózatokat.

A Rolitron komoly tapasztalatokat szerzett a hálózatépítés területén. A Cabletronnal kötött VAR-szerződés értelmében elsősorban az Ethernet, Token Ring és Apple Talk lokális hálózatok építéséhez szükséges elemek teljes választékával és értékhozóadó tevékenységével (oktatás, műszaki támogatás) igyekszik valóban működő hálózatokat kiépíteni.



Szigorúan ellenőrzött komputerek...

A hazai számítástechnika fejlődésére erőteljesen rányomta a bélyegét, hogy az igazán korszerű, nagy tudású gépek vásárlásakor meg kellett várni a COCOM-engedélyt. Ez gyakran akár fél évbe is beletelt. A reménybeli felhasználó és a hazai forgalmazó egyaránt türelmetlenül várta az engedélyt. Tudjuk, hogy sokszor ugyanaz a gép vándorolt kiállításról kiállításra, termékmutatóra. S mivel csak egyetlen darab ilyen számítógéptípus volt, a jó paraméterekkel rendelkező gép hazai piaci bevezetésénél ez az eljárás nem jelentett előnyt.

Ezen a (magyar felhasználónak) áldatlannak tűnő helyzeten segített 1990-ben egy kormányrendelet, amely lehetővé tette, hogy állami garancia segítségével, leegyszerűsített módon kerüljenek a fejlett számítástechnikai eszközök Magyarországra. Így például az IBM az első között kapott exportengedélyt Magyarországra a lebegőpontos processzoros gépek, RISC-rendszerek, adatkódoló berendezések, valamint ezek operációs rendszereinek és felhasználói szoftvereinek importjára.

A helyzet tavasz óta tovább javult. Ugyanis az IBM Magyarország Kft. a saját és az IBM világszervezetének garanciájával — a superkomputer és a kriptográfiai termékek kivételével — viszonteladói keretengedélyre terhére bármilyen számítástechnikai eszközt szabadon importálhat, betartva a behozatali eljárásokat.

A magyarországi IBM naprakész információkkal rendelkezik a belföldön eladott, üzembe helyezett termékekről és technológiákról. Ugyanis az ellenőrzött termékek nyomkövetését a COCOM-tagországok továbbra is evidenciában tartják. További könnyítés, hogy a supergépek és a kriptográfiai eszközök esetében — a vevő nevében — a magyarországi IBM kéri meg az egyedi engedélyt az exportáló országtól.

Az új helyzet kínálta lehetőségek a továbbiakban nem korlátozzák a legkorszerűbb technológia magyarországi beáramlását, amely remélhetőleg kedvező hatással lesz a hazai számítógéppark fejlesztésére.

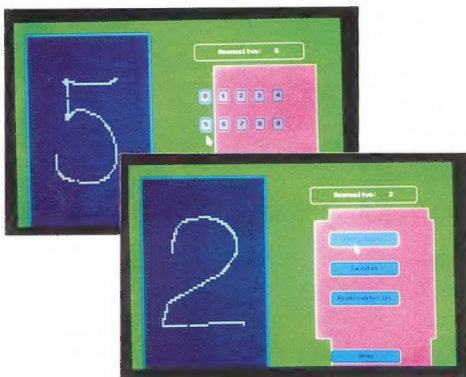
Hogyan szimuláljunk? — Neuronhálózattal!

Az emberi agy felépítését és működését utánozó neuronhálózatok olyankor segíthetnek, amikor a feladatok megoldásának pontos algoritmusát nagyon nehéz vagy lehetetlen meghatározni. Ilyen alkalmazási terület például a beszéd- és alakfelismerés vagy a képfeldolgozás. A mesterséges neuronhálózatok — az emberhez hasonlóan — minták alapján tanulnak, ezeket memorizálják, sőt a minták alapján még általánosítani is tudnak.

Alkalmazásuk területén néhány új eredmény született szinte a nagyvilágban. Neuronhálózat dolgozik a manapság oly népszerű pen-computerben a ceruzával képernyőre írt szöveg értelmezésekor; a robotok, amelyek kezelőjük szavát megértik és végrehajtják, szintén ezen az elven „gondolkodnak”. Ugyancsak neuronhálózat működik abban a repülőtéren biztonsági berendezésben, amely megtalálja a csomagokba rejtett robbanóanyagot, vagy — a repülés világánál maradván — a radar- és infravörös képek alapján a repülőgépet

tetszőleges helyzetből és irányból felismerő berendezés leseből is.

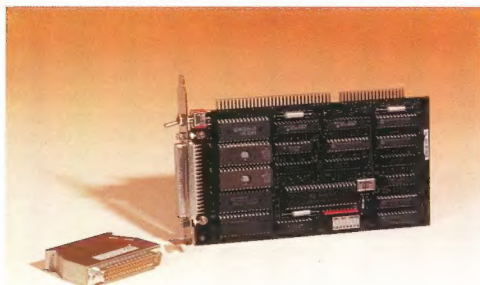
Az eddig használatos neuronhálózati modellek rendkívül lassan tanulnak. A Gerenia elnevezésű neuronhálózat-szimulátort magyar szoftverek fejlesztették ki. A hagyományos (back propagation) algoritmus helyett a program egy teljesen új elven (Classitron) alapul. Így a tanulási idő 2-3 nagyságrenddel(!) csökken. Például a kézzel írt számok felismerése 3 nap helyett 10 perc alatt beépül a „mesterséges agyba”. Ugyanez kézzel írt betűk esetében 2 hét helyett mindössze fél órát vesz igénybe. Így olyan alkalmazások is megvalósíthatók, amelyek a rendkívül hosszú tanulási idő miatt eddig szóba sem jöhettek. Hagyományosan — a kellő gyorsaság elérése érdekében — a tanításhoz mindenképp célszerű nagy- vagy szuperszámítógépet használni, még ha az applikáció PC-n is fut majd, míg a Classitron módszerrel a tanításhoz is elég a PC.



A Gerenia olyan fejlesztőrendszer, amellyel rendkívül gyorsan készíthetők alkalmazások. Az Aktív Rekord Bt. az Ifabon mutatott be egy alkalmazást (kézzel írt számok felismerését programmal); ez 48 óra alatt készült el, beleértve a hálózat betanítását is. (A demóverzió a lemez mellékletben megtalálható.) Ugyanez a feladat hagyományosan hetek-hónapok munkáját jelentette volna. Nem véletlen, hogy a külföldi szaksajtó (Computer Product News) is felfigyelt a magyar szerzőcsozra, és májusi számában már beszámolt a szakirodalomban jelenleg egyedülálló gyorsaságú neuronhálózatos rendszerről. S ha hozzátesszük, hogy a Turbo Pascalban írt fejlesztőrendszer nem is drága (99 000 Ft), valamint hogy az alkalmazások készítése rendkívül gyorsan lerövidül, akkor nem kérdéselhető meg az állítás: a már említett felhasználási területek mellett például az EKG-analízisnél vagy pénzügyi, gazdasági előrejelzéseknél lehetne okosan használni a mesterséges intelligencia e hazai gyöngyszemét.

Hogyan védekezzünk? — Titkosan!

Mindenki, aki számítógépen dolgozik, valósággal retteg attól, hogy adatai illetéktelen kezekbe kerülhetnek. Hiszen — nagy



általánosságban — jelenleg bárki bejuthat egy hálózatba egy PC-n keresztül, s ott nemcsak a vírusok terjedését idézheti elő, hanem mások félteve őrzött adatai között is szabadon garázdálkodhat. De ha az előre megfontolt szándékkal kalózkodó „okos számítástechnikusokat” figyelmen kívül is hagyjuk, gondolni kell azokra a nemtörődöm laikusokra is, akik gondatlanságból például kidobálják a bérlístáka, s ezeket a későbbiekben bárki tanulmányozhatja. Emlékezzünk csak arra, milyen sokszor csomagoltak a piacon valami portékát olyan papírba (listába), amelyen egy vállalat dolgozóinak pontos címe, összes adata és fizetése szerepelt. Ugyancsak problémák forrása lehet például, hogy rossz adatátviteli környezetben a faxszolgáltatások is illetéktelen kezekbe kerülhetnek.

Adatvédelmi megoldásra egymástól függetlenül két cég is kínál hasonló, de más fajsúlyú terméket. Mindkét társaság olyan kártyát fejlesztett ki, amely — külön-külön, az adó és a vevő gépébe helyezve — a számítógépben és a hálózatokban őrzött adatok és információk magas szintű védelmet biztosítja. A kártyával rejtjelezzik a diszken tárolt adatokat, visszakereséskor pedig időlassulás nélkül dekódolják azokat. Így a felhasználó ebből a procedurából semmit sem vesz észre. Minden kezelőnél más és más a rejtjelezési kulcs, így mindenki csak a maga által tárolt adatokhoz férhet hozzá. De természetesen vannak olyan általános kulcsok is, amelyek egy bizonyos adatkörre vonatkozó információkat rejtjeleznek. A kriptográfiai rendszer felhasználói így biztosak lehetnek abban, hogy a titkosan tárolt vagy közvetített üzeneteket mások nem tudják megfejteni, hamisítani, illetve manipulálni.

Az ITEA Kft. által fejlesztett rendszerben egy komplett szett kell ahhoz, hogy valaki hozzáférjen a számítógéphez. Mint hardver, ez egy kriptográfiai alapú kártya (CryptoPCard), s a központi kulcsellátó berendezés, valamint egy külső eszköz jelenlétét feltételezi. A CryptoPCard tulajdonképpen szabványos AT bővítmőkártya, amelyet a központi kulcsellátó berendezés lát el rejtjelezési kulcsokkal. A kulcsgenerálást a felhasználó maga is elvégezheti, mivel a CryptoPCard az alkalmazó saját rejtjelezési algoritmusát is értelmezi. A kulcs és az algoritmus egy része a külső eszközben található. A rejtjelezés visszafejtésére nincs mód, hiszen ehhez valamennyi fizikai és logikai elemet kellene egyidejűleg ismerni.

A rendszerben az univerzálisan kialakított moduláris szoftver a felhasználói programokkal tehető egyedivé. Adatvédelmi rendszer alkalmazása esetén a felhasználói programokat és az adatbázisokat kriptográfiai szempontból is meg kell tervezni. Ilyenkor kell dönteni a használandó kulcsrendszeréről, a hozzáférési hierarchiáról, a fájl- vagy mezőszintű

titkosításról és a rejtjelezési algoritmusáról. De nem okoz gondot az sem, hogy a már működő programokat kriptoparancsokkal egészítsük ki. Az államigazgatásban, az önkormányzatoknál, a banki szférában, a biztosítóknál és az egészségügyben kiütően használható kártya 45 000 és 79 000 Ft közötti (kiepítéstől függő) árával méltányos beruházás az adatok védelme érdekében.

Az olcsóbb árfekvésű, de felhasználóbarát és már XT-ben is használható — az előbbivel ennyiben konkuráló — kártyát (BIS SEC 1.0) a Bis Kft. fejlesztette ki. Ez néhány hasonló szolgáltatásával 50 000 forintért kínál alternatívát a felhasználók egy részének.

A rejtjelezés bájtól bajra halad, az adatok a processzorban nem olvashatók ki, a kódolás-dekódolás kívülről nem követhető. A fájl- és adatrejtjelezést megvalósító kártyán valódi véletlenszám-generátor van. Fontos, hogy rendszeresen ellenőrizik az öntesztelés működését. A hálózatban is működő rendszerben a beépített hardverérzékelő (egy floppy) erősti a PC fizikai hozzáférés-védelmet. Ha kivesszük a 3,5"-os floppyt a gépből, akkor elsőtétül a képernyő, amellyel a hozzáférést és a számítógépet egyaránt védjük. Az adatvédelmi rendszer kezelése kényelmes, hiszen a 3,5"-os floppyval ugyanazt csinálhatjuk meg, mint jelszóval és egyéb segédeszközökkel.

Az adatvédelem egyre növekvő fontosságát jól jelzi, hogy az Országgyűlés is tárgyalja az adatvédelmi törvényt. A „klasszikus” és a profi felhasználók kellemes helyzetben vannak, mert eldönthetik, hogy a különböző „védekezési módok” közül melyiket választják.

Sziebig Andrea

SHARP

Másológép centrum

Példa árakból:

SHARP Z 30 39 900,- + áfa

SF 6100 74 900,- + áfa

SF 7370 113 900,- + áfa

SF 7850 173 900,- + áfa

Minden géphez indulókészlet: 15 900,- + áfa

- Másológépek forgalmazása és szakszervize.

- Kellék-, festék-, papírellátás.

Canon, Mita, Ricoh, Toshiba másológépekhez
festékek és kellek DISZKONT ÁRON.



SF-6100

E·Copy Kft.

Fénymásolók értékesítése
és szakszervize

1146 Bp. XIV., Thököly út. 57/B

Tel.: 251-1869, 252-2566



A NASHUA 3000-es analóg másológép-család

- 3115, 3018, 3120, 3133, 3133 LD
- 15, 18, 20, 33 percenkénti másolatszám
- 50-200% zoom
- Méretre másolás
- A5-A3
- Opciók: Szorter, szorter + tűző
- Automata behúzó
- Editálótábla
- Kétszínű másolás
- Kétoldalas másolás

A NASHUA 300-as digitális sokszorosítógép-család

- CP 305, CP 310, CP 330, CP 375
- 300, ill. 400 DPI
- A4, B4, A3 max. nyomófelület,
nagyítás-kicsinyítés
- Változtatható sebesség 40-120 másolat/perc
- Papírigény 50-215 g/nm között
- Tiszta környezetbarát technológia
- Kis helyigény
- Opciók: Szorter
- Számítógéppinterface
- Többszínű másolás
- Képszerkesztés



Nashua

Magyarországi vezérképvislet

1138 Budapest XIII., Váci út 168. Tel.: 149-7520 • Fax: 129-0769



ComputerLand®

1055 BUDAPEST BALASSI BÁLINT U. 7.
TELEFON: 269-0171 • FAX: 269-0178